取扱説明書

ディジタルストレージスコープ

DS-9121/9122

リモートインターフェース編

IWATSU

はじめに

◇この度は岩通の電子測定器をお買い上げいただき、ありがとうございます。今後とも岩通の電子測定器を末長くご愛用いただきますよう、お願い申し上げます。
◇本取扱説明書をよくお読みの上、内容を理解してからお使いください。お読みになった後も、大切に保管してください。

安全にご使用いただくために

本製品を安全にお使いいただき、人体への危害や財産への損害を未然に防ぐために守っていただきたい事項が取扱説明編の「⚠警告」と「⚠注意」に記載されています。必ずお読みください。更に、パネルに注意を促す記号が記されています。

取扱説明編の「⚠警告」と「⚠注意」の説明

⚠ 警 告	ここに記載されている事項を無視して、誤った取り扱いをすると、人が死亡する または 重傷を負う可能性が想定されます。
⚠ 注 意	ここに記載されている事項を無視して、誤った取り扱いをすると、人が傷害を負う または 機器が破損する可能性が想定されます。

パネルの記号の説明

⚠ 警告記号	人体を保護する および 本器を損傷から守るため、取扱説明書の記載事項を参照の上、ご使用いただくための記号です。

ご注意

◇本取扱説明書の内容を性能・機能の向上などにより一部を予告なく変更することがあります。
◇本取扱説明書の内容を無断で転載、複製することを禁止します。
◇本製品に対するお問い合わせなどがございましたら、支店・支部・営業部・TA センタなどにご連絡ください（巻末の『ネットワーク』参照）。

履　歴

◇1993年 8月 第 1 版発行
◇1995年 8月 第 4 版発行

目　　次

はじめに …… I
安全にご使用いただくために …… I

第1部 GP-IB インタフェース …… 1- 1
1.1 概　要 …… 1- 2
1.2 仕　様 …… 1- 2
1.3 接　続 …… 1- 3
1.4 機器アドレス …… 1- 3
1.5 状態表示 LED …… 1- 3
1.6 インタフェースメッセージに対する応答 …… 1- 4
1.6.1 リモート／ローカル制御 …… 1- 4
1.6.2 GET (Group Execute Trigger) …… 1- 5
1.6.3 DCL・SDC …… 1- 5
1.7 メッセージ …… 1- 5
1.7.1 メッセージフォーマット …… 1- 6
1.7.2 マルチコマンド …… 1- 6
1.7.3 クエリ …… 1- 7
1.7.4 ヘッダ部ニーモック …… 1- 8
1.7.5 データ部フォーマット …… 1- 8
1.7.6 デリミタ (<PROGRAM/RESPONSE MESSAGE TERMINATOR>) …… 1-10
1.7.7 入出力バッファ …… 1-10
1.8 サービス要求とステータスレポート構造 …… 1-11
1.8.1 サービス要求 (SRQ) …… 1-11
1.8.2 ステータス バイト レジスタ …… 1-12
1.8.3 サービス リクエスト イネイブル レジスタ …… 1-12
1.8.4 スタンダード イベント ステータス レジスタ …… 1-13
1.8.5 スタンダード イベント ステータス イネイブル レジスタ …… 1-13
1.8.6 出力キュー …… 1-13
1.8.7 GO／NOGO イベント ステータス レジスタ …… 1-13
1.8.8 GO／NOGO イベント ステータス イネイブル レジスタ …… 1-14
1.8.9 トリガ イベント ステータス レジスタ …… 1-14
1.8.10 トリガ イベント ステータス イネイブル レジスタ …… 1-14
1.8.11 ローカル キュー …… 1-14
1.8.12 エラー キュー …… 1-14

第2部 RS-232C インタフェース …… 2- 1
2.1 概　要 …… 2- 2
2.2 仕　様 …… 2- 2
2.3 設定条件 …… 2- 3
2.4 信号ラインとコネクタピン …… 2- 3
2.5 コネクタとケーブル …… 2- 4
2.6 コントローラによる制御 …… 2- 4
2.6.1 コマンド …… 2- 4
2.6.2 コントローラ側の処理 …… 2- 5
2.7 PC-9801との接続 …… 2- 6

第３部　機器固有コマンド …… 3- 1
機器固有コマンド …… 3- 2
コマンドリスト …… 3- 3
3.1 共通コマンド …… 3- 8
3.1.1 *IDN? クエリ(Identification Number) …… 3- 9
3.1.2 *OPT? クエリ(Option) …… 3-10
3.1.3 *LRN? クエリ(Learn) …… 3-11
3.1.4 *RST コマンド(Reset) …… 3-11
3.1.5 *TST? クエリ(Test) …… 3-11
3.1.6 *OPC コマンド/ *OPC? クエリ(Operation Complete) …… 3-12
3.1.7 *WAI コマンド(Wait) …… 3-12
3.1.8 *CLS コマンド(Clear Status) …… 3-12
3.1.9 *ESE コマンド/ *ESE? クエリ(Event Status Enable) …… 3-13
3.1.10 *ESR? クエリ(Event Status Register) …… 3-14
3.1.11 *PSC コマンド/ *PSC? クエリ(Power on Status Clear) …… 3-15
3.1.12 *SRE コマンド/ *SRE? クエリ(Service Request Enable) …… 3-16
3.1.13 *STB? クエリ(Status Byte) …… 3-17
3.1.14 *TRG コマンド(Trigger) …… 3-18
3.1.15 *RCL コマンド(Recall) …… 3-18
3.1.16 *SAV コマンド(Save) …… 3-19
3.2 ルート (ROOT) レベルコマンド …… 3-20
3.2.1 RUN コマンド …… 3-21
3.2.2 STOP コマンド …… 3-21
3.2.3 CLEer コマンド …… 3-21
3.2.4 ASCHannel コマンド/ ASCHannel? クエリ …… 3-22
3.2.5 ASETup コマンド …… 3-22
3.2.6 WAIT コマンド …… 3-23
3.2.7 WSGL コマンド …… 3-23
3.2.8 TESR? クエリ …… 3-23
3.2.9 TESE コマンド/ TESE? クエリ …… 3-24
3.2.10 LCLQ? クエリ …… 3-24
3.3 ACQuire 関連コマンド …… 3-25
3.3.1 MEMory コマンド/ MEMory? クエリ …… 3-26
3.3.2 POINts コマンド/ POINts? クエリ …… 3-27
3.3.3 TYPE コマンド/ TYPE? クエリ …… 3-28
3.3.4 COUNts コマンド/ COUNts? クエリ …… 3-29
3.3.5 INTerpolate コマンド/ INTerpolate? クエリ …… 3-30
3.3.6 RPAGe コマンド/ RPAGe? クエリ …… 3-31
3.3.7 DPAGe コマンド/ DPAGe? クエリ …… 3-32
3.3.8 RMODe コマンド／RMODe? クリエ …… 3-33
3.3.9 SMOothing／SMOothing? …… 3-34
3.4 TIMebase 関連コマンド …… 3-36
3.4.1 SAMPling コマンド／ SAMPling? クエリ …… 3-37
3.4.2 SDIV コマンド/ SDIV? クエリ …… 3-38
3.4.3 SCRoll コマンド/ SCRoll? クエリ …… 3-39
3.4.4 BSWeep コマンド/ BSWeep? クエリ …… 3-40
3.4.5 DISPlay コマンド/ DISPlay? クエリ …… 3-41
3.4.6 BSDiv コマンド/ BSDiv? クエリ …… 3-41
3.4.7 DELay コマンド/ DELay? クエリ …… 3-43
3.4.8 TSEParation コマンド／ TSEParation? クエリ …… 3-44

3.5 CHannel 関連コマンド 3-45
3.5.1 VDIV コマンド/ VDIV? クエリ 3- 47
3.5.2 ZOOM コマンド/ ZOOM? クエリ 3- 49
3.5.3 COUPling コマンド/ COUPling? クエリ 3- 50
3.5.4 POSition コマンド/ POSition? クエリ 3- 51
3.5.5 TRACe コマンド/ TRACe? クエリ 3- 52
3.5.6 INVert コマンド/ INVert? クエリ 3- 53
3.5.7 FILTer コマンド/ FILTer? クエリ 3- 54
3.5.8 PROBe コマンド/ PROBe? クエリ 3- 56
3.6 TRIGger 関連コマンド 3- 58
3.6.1 MODE コマンド/ MODE? クエリ 3- 60
3.6.2 SWEep コマンド/ SWEep? クエリ 3- 61
3.6.3 SOURce コマンド/ SOURce? クエリ 3- 62
3.6.4 PATH コマンド/ PATH? クエリ 3- 63
3.6.5 LEVel コマンド/ LEVel? クエリ 3- 64
3.6.6 SLOPe コマンド/ SLOPe? クエリ 3- 65
3.6.7 COUPling コマンド/ COUPling? クエリ 3- 66
3.6.8 REJect コマンド/ REJect? クエリ 3- 67
3.6.9 NREJect コマンド/ NREJect? クエリ 3- 68
3.6.10 POSition コマンド/ POSition? クエリ 3- 69
3.6.11 HOLDoff コマンド/ HOLDoff? クエリ 3- 70
3.6.12 イベントトリガ専用コマンド 3- 71
3.6.13 コンビネーショントリガ専用コマンド 3- 74
3.6.14 TV／HDTV TRIGGER 専用コマンド 3- 77
3.7 ARM 関連コマンド 3- 80
3.7.1 SOURce コマンド/ SOURce? クエリ 3- 81
3.7.2 LEVel コマンド/ LEVel? クエリ 3- 82
3.7.3 SLOPe コマンド/ SLOPe? クエリ 3- 84
3.7.4 COUPling コマンド/ COUPling? クエリ 3- 85
3.7.5 REJect コマンド/ REJect? クエリ 3- 86
3.7.6 NREJect コマンド/ NREJect? クエリ 3- 87
3.8 DISPlay 関連コマンド 3- 88
3.8.1 SPLit コマンド/ SPLit? クエリ 3- 90
3.8.2 GRATicule コマンド/ GRATicule? クエリ 3- 91
3.8.3 MODE コマンド/ MODE? クエリ 3- 92
3.8.4 PERSistence コマンド/ PERSistence? クエリ 3- 93
3.8.5 TEXT コマンド 3- 94
3.8.6 XY<x> コマンド/ XY<x>? クエリ 3- 96
3.8.7 CALCulate コマンド/ CALCulate? クエリ 3- 98
3.8.8 DATE コマンド/ DATE? クエリ 3-102
3.8.9 RGB コマンド/ RGB? クエリ 3-103
3.9 REFerence 関連コマンド 3-104
3.9.1 DISPlay コマンド/ DISPlay? クエリ 3-105
3.9.2 XPOSition コマンド/ XPOSision? クエリ 3-106
3.9.3 YPOSition コマンド/ YPOSition? クエリ 3-107
3.9.4 SET コマンド 3-108
3.9.5 DATA コマンド/ DATA? クエリ 3-109

3.10 CURSor 関連コマンド …… 3-110
3.10.1 TMOVe コマンド/ TMOVe? クエリ …… 3-112
3.10.2 TPReference コマンド/ TPReference? クエリ …… 3-113
3.10.3 TPRSet コマンド …… 3-114
3.10.4 TPMeasure コマンド/ TPMeasure? クエリ …… 3-114
3.10.5 TFUNction コマンド/ TFUNction? クエリ …… 3-115
3.10.6 TSOurce コマンド/ TSOurce? クエリ …… 3-116
3.10.7 TRENtry コマンド …… 3-116
3.10.8 TDELta? クエリ …… 3-117
3.10.9 VPReference コマンド/ VPReference? クエリ …… 3-118
3.10.10 VPMeasure コマンド/ VPMeasure? クエリ …… 3-119
3.10.11 VFUNction コマンド/ VFUNction? クエリ …… 3-120
3.10.12 VSOurce コマンド/ VSOurce? クエリ …… 3-121
3.10.13 VRENtry コマンド …… 3-122
3.10.14 VDELta? クエリ …… 3-122
3.10.15 SPReference コマンド／SPRefrence? クエリ …… 3-123
3.10.16 SPMeasure コマンド／SPMeasure? クエリ …… 3-124
3.10.17 SFUNction コマンド／SFUNction? クエリ …… 3-125
3.10.18 SDELta クエリ …… 3-126
3.11 COPY コマンド …… 3-127
3.11.1 DEVice コマンド/ DEVice? クエリ …… 3-128
3.11.2 ROLL コマンド/ ROLL? クエリ …… 3-129
3.11.3 FEED コマンド …… 3-130
3.11.4 EXECute コマンド …… 3-130
3.12 PLOT コマンド …… 3-131
3.12.1 P1 コマンド/ P1? クエリ …… 3-132
3.12.2 P2 コマンド/ P2? クエリ …… 3-133
3.12.3 ROTation コマンド/ ROTation? クエリ …… 3-134
3.12.4 PORT コマンド/ PORT? クエリ …… 3-135
3.12.5 CURSor コマンド/ CURSor? クエリ …… 3-136
3.12.6 READout コマンド/ READout? クエリ …… 3-137
3.12.7 GRATicule コマンド/ GRATicule? クエリ …… 3-138
3.12.8 PEN コマンド/ PEN? クエリ …… 3-139
3.12.9 DUMMy コマンド …… 3-139
3.13 SYSTem 関連コマンド …… 3-140
3.13.1 DATE コマンド/ DATE? クエリ …… 3-141
3.13.2 TIME コマンド/ TIME? クエリ(SCPI) …… 3-142
3.13.3 HEADer コマンド/ HEADer? クエリ …… 3-143
3.13.4 LONGform コマンド/ LONGform? クエリ …… 3-144
3.13.5 AVERage コマンド/ AVERage? クエリ …… 3-145
3.13.6 ERRor? クエリ …… 3-145
3.13.7 SET コマンド/ SET? クエリ …… 3-146
3.13.8 BALance コマンド …… 3-147
3.13.9 GAIN コマンド …… 3-147
3.13.10 ADC コマンド …… 3-147
3.13.11 TRIG コマンド …… 3-148
3.13.12 SPEed コマンド／SPEed? クエリ …… 3-148

3.14 SETup 関連コマンド …… 3-149
3.14.1 SAVE コマンド …… 3-149
3.14.2 RECall コマンド …… 3-150
3.14.3 DELete コマンド …… 3-151
3.14.4 DIRectry? クエリ …… 3-152
3.14.5 INITialize コマンド …… 3-154
3.14.6 SECurity コマンド/ SECurity? クエリ …… 3-155
3.15 MCARd 関連コマンド …… 3-156
3.15.1 CDIRectry コマンド/ CDIRectry? クエリ …… 3-158
3.15.2 SAVE コマンド …… 3-159
3.15.3 RECall コマンド …… 3-160
3.15.4 DELete コマンド …… 3-161
3.15.5 DIRectry? クエリ …… 3-162
3.15.6 INITialize コマンド …… 3-164
3.15.7 SOURce コマンド/ SOURce? クエリ …… 3-165
3.15.8 DESTination コマンド/ DESTination? クエリ …… 3-166
3.15.9 READ? クエリ …… 3-167
3.15.10 WRITe コマンド …… 3-168
3.15.11 SECurity コマンド/ SECurity? クエリ …… 3-169
3.16 WAVeform (波形データ) 関連コマンド …… 3-170
3.16.1 FORMat コマンド/ FORMat? クエリ …… 3-171
3.16.2 BORDer コマンド/ BORDer? クエリ …… 3-172
3.16.3 SOURce コマンド/ SOURce? クエリ …… 3-173
3.16.4 POINts コマンド/ POINts? クエリ …… 3-174
3.16.5 STARt コマンド/ STARt? クエリ …… 3-175
3.16.6 DATA? クエリ …… 3-176
3.16.7 TYPE? クエリ …… 3-178
3.16.8 XSIZe? クエリ …… 3-178
3.16.9 XSCale? クエリ …… 3-179
3.16.10 YSCale? クエリ …… 3-179
3.16.11 XOFFset? クエリ …… 3-180
3.16.12 YOFFset? クエリ …… 3-180

第4部 プログラム例 …… 4- 1
4.1 GP-IB …… 4- 2
4.2 RS-232C …… 4- 8

図のリスト

図3.1.1　共通コマンド全体像 …… 3- 8
図3.2.1　ルートレベルコマンド全体像 …… 3- 20
図3.3.1　ACQuire 関連コマンド全体像 …… 3- 25
図3.4.1　TIMebase 関連コマンド全体像 …… 3- 36
図3.5.1　CHannel 関連コマンド全体像 …… 3- 46
図3.6.1　TRIGger 関連コマンド全体像 …… 3- 58
図3.7.1　ARM 関連コマンド全体像 …… 3- 80
図3.8.1　DISPlay 関連コマンド全体像 …… 3- 88
図3.9.1　REFerence 関連コマンド全体像 …… 3-104
図3.10.1　CURSor 関連コマンド全体像 …… 3-110
図3.11.1　COPY 関連コマンド全体像 …… 3-127
図3.12.1　:PLOT 関連コマンド全体像 …… 3-131
図3.13.1　SYSTem 関連コマンド全体像 …… 3-140
図3.14.1　SETup 関連コマンド全体像 …… 3-149
図3.15.1　MCARd 関連コマンド全体像 …… 3-157
図3.16.1　WAVeform 関連コマンド全体像 …… 3-170

表のリスト

表1.2.1　GP-IB インタフェース機能 …… 1- 2
表1.7.1　サフィックス乗数を表すニーモニック …… 1- 8
表2.4.1　信号ラインとコネクタピン …… 2- 3
表2.6.1　RS-232C 専用コマンド …… 2- 4
表3.1　コマンドリスト(1/5) …… 3- 3
表3.2　コマンドリスト(2/5) …… 3- 4
表3.3　コマンドリスト(3/5) …… 3- 5
表3.4　コマンドリスト(4/5) …… 3- 6
表3.5　コマンドリスト(5/5) …… 3- 7
表3.1.1　スタンダードイベントステータスイネイブルレジスタ構成 …… 3- 13
表3.1.2　スタンダードイベントステータスレジスタ構成 …… 3- 14
表3.1.3　サービスリクエストイネイブルレジスタ構成 …… 3- 16
表3.1.4　ステータスバイトレジスタ構成 …… 3- 17

Memo

第１部　ＧＰ－ＩＢインタフェース

1.1 概　要

本器は GP-IB インタフェースを標準装備しています。 GP-IB コントローラ機能を持つパーソナルコンピュータなどに接続すると、自動計測システムを容易に構成することができます。
本器のほとんどの機能をリモート制御することができますが、次の項目はリモート制御できません。

- 電源スイッチの ON/OFF
- CRT の INTEN,CONTRAST ボリューム
- SYSTEM メニュー内の CAL & TEST サブ メニュー関連の機能
- SYSTEM メニュー内の REMOTE I/F サブ メニュー関連の機能
- DS-900 内蔵プリンタ(オプション)の PAPER HOLD/FREE レバーの開閉
- ソフトウェア オプション プログラムのダウン ロード関連の機能など、リモート制御可能な機能と制御コマンドなどの詳細に関しては、"第3部 機器固有コマンド"の項をご参照下さい。

注　　意

- 本器は RS-232C インタフェースによるリモート制御も可能ですが、既に GP-IB リモート状態になっている場合、RS-232C インタフェースによるリモート制御はできません。逆に、RS-232C リモート状態になっている場合、GP-IB インタフェースによるリモート制御はできません。
- RS-232C インタフェースはオプションです。

1.2 仕　様

a.性　能

電気的･機械的･機能的仕様は IEEE Std.488.1-1987 および JIS C 1901-1987 に準拠します。
コマンド フォーマット プロトコルなどに関する仕様は IEEE Std.488.2-1987 に準拠します。

b.インタフェース機能

本器は GP-IB に対して 表 1.2.1 に示す IEEE488.1 のサブセットを持ちます。

表 1.2.1 GP-IB インタフェース機能

ｻﾌﾞｾｯﾄ	機　能　の　内　容
ＳＨ１	ソースハンドシェークの全機能を持つ
ＡＨ１	アクセプタハンドシェークの全機能を持つ
Ｔ５	基本的トーカ、シリアルポール、MLA によるトーカ指定解除 および トークオンリ機能を持つ
ＴＥＯ	拡張トーカの機能を持たない
Ｌ４	基本的リスナ および MTA によるリスナ指定解除機能を持つ
ＬＥＯ	拡張リスナの機能を持たない
ＳＲ１	サービスリクエストの全機能を持つ
ＲＬ１	リモートローカルの全機能を持つ
ＰＰＯ	パラレルポールの機能を持たない
ＤＣ１	デバイスクリアの全機能を持つ
ＤＴ１	デバイストリガの全機能を持つ
ＣＯ	コントローラの機能を持たない
Ｅ２	３ステート ドライバ使用

1.3 接　続

注　　意

・ケーブルの取り付け または 取り外しを行うときは、GP-IB コネクタに接続しているすべての機器の電源を OFF にして下さい。
・GP-IB システムを動作させる場合には、システムバスに接続されたすべての機器の電源を ON にして下さい。

a.1 つのシステムに接続可能な GP-IB デバイスは最大 15 台です。
b.システム内の各デバイスを互いに接続するために使用できるケーブルの合計長は、20m 以下でかつ 2m×(システム構成デバイスの総数)以下です。
c.システム内のケーブル長の配分はシステム構築者が自由に行うことができます。但し、デバイス間のケーブル長が 4m 以上になる部分がある場合には、ノイズマージン等に充分注意してください。
d.ケーブルの接続方法(スター、ディージーチェーンなど)も使用者が適当と判断する方法で行うことができます。但し､グランドループができるような接続はしないでください。
e.本器背面のコネクタへスタックするケーブルの段数は、機械的強度の関係により、3 段以下としてください。
f.接続ケーブルは、IEEE 488.1 または JIS C 1901 規格に適合するものを使用します。システムの信頼性向上のため、EMC 対策品(金属ハウジング使用のコネクタ)の採用を推奨します。
g.GP-IB システムは電気的･機械的に比較的良好な環境での使用に適します。
h.システム構築にあたっては JIS C 1901-1987 第 6 節および附属書 J を参考にして下さい。

1.4 機器アドレス

a.リスナアドレス･トーカアドレスとして、0～30 の範囲で任意のアドレスを選択できます。
b.本器のリスナアドレスとトーカアドレス(コードセット下位 5 ビット)は共通です。
　リスナアドレスとして有効なコードセット：コード表 2 列 0～3 列 14。
　トーカアドレスとして有効なコードセット：コード表 4 列 0～5 列 14。
c.工場出荷時の設定アドレスは 10。
　MLA：コード表 2 列 10。　MTA：コード表 4 列 10。

アドレスの設定はシステム メニュー内 REMOTE I/F SUB MENU で行います。第9部の"9.1.2 機器アドレス､ボーレート"をご参照下さい。

1.5 状態表示 LED

本器の正面パネル中央上部には GP-IB の状態表示用 LED が 4 個ついています。
a.REMOTE
　本器がリモート状態のときに点灯します。
　RS-232C インタフェースによってリモート制御されている場合にも点灯しますので、注意してください。
b.TALK
　本器がトーカとして指定されている期間点灯します。
c.LISTEN
　本器がリスナとして指定されている期間点灯します。
d.SRQ
　本器がサービスリクエストの要求元である期間点灯します。

1.6 インタフェースメッセージに対する応答

1.6.1 リモート/ローカル制御

本器の各機能をインタフェース経由のメッセージでリモートコントロールすることができます。
インタフェース経由のメッセージによってコントロールされる状態をリモートモードと呼びます。
これに対して、パネルよりのキー入力によって動作している状態をローカルモードと呼びます。
本器は IEEE Std.488.1-1987 および JIS C 1901-1987 で規定されたリモート/ローカルの全機能を持っています。リモート/ローカルの状態遷移などの詳細については規格書の RL 機能の項をご参照ください。

a.ローカルモード

- ローカルモードではパネルからのキー操作によって本器の動作をコントロールします。
- REMOTE LED は消灯しています。
- 本器は電源投入によりローカルモードとなります。
- リモートモードで動作している本器をローカルモードに戻す方法は 4 種類があります。
  1)電源の再投入を行う。
  2)REN ラインを偽(電気的には HIGH レベル)にする。
  3)[LOCAL(SYSTEM)]ｷｰを押す。
  4)本器をリスナに指定して、アドレスコマンド GTL を送る。
- リモートモードからローカルモードに戻った場合、リモートモードで行われた条件設定(SEC/DIV,VOLTS/DIV などの各種セットアップ)はそのまま引き継がれます。ただし、コマンド解釈の途中でローカルモードに移行すると、解釈の完了していないコマンドは無視されます。

b.リモートモード

- GP-IB インタフェース経由のメッセージで本器の動作をコントロールします。
- パネルからのキー入力は[LOCAL(SYSTEM)]を除いて、総て無視されます。
- 本器の動作モードをローカルからリモートに切り換えるには、REN ラインが真(電気的に LOW レベル)の状態で、本器をリスナに指定(MLA 受信)する必要があります。
- リモートモードに移行すると、CRT 表示は SHOW 表示画面となります。
- 条件設定(SEC/DIV,VOLTS/DIV などの各種セットアップ)は、ローカルモードでの設定をそのまま引き継ぎます。
- リモート期間中は、REMOTE LED が点灯します。

c.ローカルロックアウトモード

- 本器はは次のいずれの場合にもローカルロックアウトモードになります。
  1)リモート状態にあるときにユニバーサルコマンド LLO を受信した場合。
  2)ユニバーサルコマンド LLO を受信した後に、リモートモードになった場合。
- ローカルロックアウトモードは、オペレータが誤って[LOCAL(SYSTEM)]ｷｰを押してもローカルモードに戻らないようにするモードです。[LOCAL(SYSTEM)]ｷｰ入力が無視される以外の動作・応答はリモートモードと同じです。
- ローカルロックアウト状態にある本器をローカルモードに戻すには、本器をリスナに指定して GTL メッセージを送信します。但し、再度リモートモードに切り換えた場合には、改めて LLO コマンドを受け取らなくても、ローカルロックアウトモードになります。
- ローカルロックアウトモードを完全に解除する(単なるリモートモードにする)には、以下のいずれかの方法により一旦ローカルモードにする必要があります。
  1)電源の再投入を行う。
  2)REN ラインを偽(電気的には HIGH レベル)にする。

1.6.2 GET(Group Execute Trigger)

・本器はリスナに指定されている状態で、アドレスコマンド GET を受信すると RUN 状態となります。
・本器が GET コマンドを受信した場合、内部処理が完了するまで、NDAC 信号をホールドすることによりハンドシェイクをホールドオフさせます。

1.6.3 DCL・SDC

・本器はユニバーサルコマンド DCL(Device Clear)を受信すると、デバイス機能の初期化を実行します。リスナに指定されている状態でアドレスコマンド SDC(Selected Device Clear)を受信した場合も同様です。
・DCL または SDC メッセージによる初期化の内容は以下のとおりです。
1)データ入出力動作の終了、及び入力バッファと出力キューをクリアします。
これにともない、ステータスバイトレジスタの MAV ビットをクリアします。この結果、MSS ビットにも影響することがあります。
2)COPY 動作を停止します。
3)TRIGGER SWEEP=SINGLE(または TRIG'D)かつ RUN 中の場合には STOP 状態にする。
・本器が DCL または SDC インタフェースメッセージを受信する場合、内部のマイクロプロセッサが受信を認識するまで、NDAC 信号をホールドすることにより、ハンドシェイクをホールドオフさせます。

1.7 メッセージ

本器がコントローラと通信するプログラムメッセージの構成や本器の送受信動作について述べます。
詳細なデータフォーマットなどについては "第3部 機器固有コマンド"の項をご参照下さい。

備　考

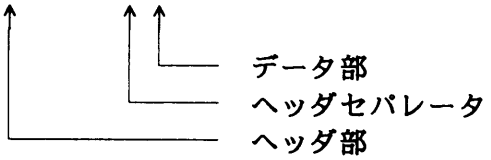
IEEE488.2規格で定められた各エレメントを以下のように表記します。

<PROGRAM MESSAGE>	→	<プログラムメッセージ>
<CHARACTER PROGRAM DATA>	→	<文字プログラムデータ>
<DECIMAL NUMERIC PROGRAM DATA>	→	<数値プログラムデータ>
<SUFFIX PROGRAM DATA>	→	サフィックス
<STRING PROGRAM DATA>	→	<文字列プログラムデータ>
<ARBITRARY BLOCK PROGRAM DATA>	→	<任意ブロックプログラムデータ>
<RESPONSE MESSAGE>	→	<応答メッセージ>
<CHARACTER RESPONSE DATA>	→	<文字応答データ>
<NR1 NUMERIC RESPONSE DATA>	→	<NR1 数値応答データ>
<NR3 NUMERIC RESPONSE DATA>	→	<NR3 数値応答データ>
<STRING RESPONSE DATA>	→	<文字列応答データ>
<DEFINITE LENGTH ARBITRARY BLOCK RESPONSE DATA>	→	<決定長任意ブロック応答データ>
<ARBITRARY ASCII RESPONSE DATA>	→	<任意 ASCII 応答データ>

1.7.1 メッセージフォーマット

1 個のメッセージユニットはヘッダ部とデータ部と両者を区切るヘッダセパレータで構成されます。

[例]　　:CHANNEL1:TRACE　ON



a.ヘッダ部

・ヘッダ部は ASCII 文字で構成されます。コロン(:)で区切られた 1 から 3 個のニーモニックから成り、本器の機能あるいは動作を表します。
・上記は本器の設定を行うためのコマンドの例です。
・設定内容を問い合わせる(本器に応答データを作成させる)クエリの場合は、ヘッダ部の最後の文字が ? マークとなります。

b.ヘッダセパレータ

・ヘッダ部とデータ部の区切りを示します。1 文字以上のスペースコード(ASCII文字)となります。IEEE488.2 規格の <white space> 文字でも構いません。
・データ部を持たない または データ部を省略する場合には、ヘッダセパレータも省略可能です。

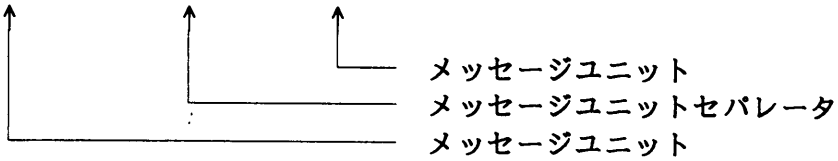
c.データ部

・データ部はヘッダ部で指定した機能の具体的な設定内容を示すパラメータです。
ニーモニック(文字列) または 数値で構成されることもあります。
バイナリコードのブロックデータになることもあります。
・複数のパラメータを必要とする場合は各々のパラメータのセパレータとしてカンマ(,)を使用します。
・パラメータの個数やその構成はコマンド毎に異なります。データ部の詳細なフォーマットについては "第3部 機器固有コマンド" の項をご参照下さい。

1.7.2　マルチコマンド

・複数のメッセージユニット集めて 1 個のメッセージとして構成することができます。
・各メッセージユニットをセミコロン(;)で区切ってつなげます。

[例]　　:CHANNEL1:TRACE　ON;:CHANNEL2:TRACE　ON



・本器の機器固有コマンドは <compound command program header> と呼ばれる階層構造のコマンドになっています。例えば CHANNEL1 という親コマンドは TRACE･VDIV･COUPLING などのサブコマンドと組み合わせて(:CHANNEL1:TRACE または :CHANNEL1:VDIV などのように)ヘッダ部が構成されます。共通の親コマンドから成るヘッダ部を持つメッセージユニットをつなげてメッセージを構成する場合、後に続くメッセージユニットでは親コマンドを省略することができます。

[例]　:CHANNEL1:TRACE　ON;:CHANNEL1:VDIV　50mV;:CHANNEL1:COUPLING　DC を
:CHANNEL1:TRACE　ON;VDIV　50mV;COUPLING　DC と書くことができます(親コマンドを省略した場合、ヘッダ部にコロン(:)をつけないことに注意)。

以下、この機能のことを「親コマンド CHANNEL1 がヘッダパスとして指定された」と表現します。

・ヘッダパス指定はそのメッセージ内(デリミタまで)でのみ有効です。次のメッセージでは、その先頭のメッセージユニットの親コマンドがヘッダパスとして指定されます。従って、メッセージの先頭ではヘッダ部の最初の文字のコロン(:)を省略しても親コマンドとみなされます。

[例] CHANNEL1:TRACE ON は
:CHANNEL1:TRACE ON と同じと解釈されます。

PRINT@DS%;":CHANNEL1:TRACE ON;VDIV 50mV" は正しいステートメントですが、
PRINT@DS%;":CHANNEL1:TRACE ON" と
PRINT@DS%;"VDIV 50mV" の 2 行に分けるとエラーとなります。本器は VDIV という親コマンドを持ちません。

・メッセージユニットの先頭がコロン(:)で始まっている場合、そのコマンドを親コマンドとみなします。ヘッダパスも更新され、そのコマンドが新しいヘッダパスとなります。

[例] :CHANNEL1:TRACE ON;VDIV 50mV;:CHANNEL2:TRACE ON;VDIV 50mV
↑ ↑
ヘッダパスは CHANNEL1 ヘッダパスは CHANNEL2

・本器はコマンドツリーをさかのぼってコマンド検索は行いません。従って、異なる親コマンドのメッセージユニットを 1 個のメッセージとする場合、ヘッダパスを切り換えるべきメッセージユニットの先頭は必ずコロン(:)で始めて下さい。

[例] :CHANNEL1:TRACE ON;VDIV 50mV;:CHANNEL2:TRACE ON;VDIV 50mV
↑
省略できない

・IEEE488.2 で規定された共通コマンド(*WAI、*OPCなど)には上記のヘッダパスに関するルールは適用されません。
共通コマンドはヘッダパスを無視して実行されます。ただし、それまでのヘッダパス指定を破壊するわけではなく、共通コマンドの次のメッセージユニットの解釈時には以前のヘッダパス指定が有効となります。

[例] :CHANNEL1:TRACE ON;*WAI;VDIV 50mV;COUPLING DC
↑ ↑
ヘッダパスは CHANNEL1
共通コマンド *WAI を正常に実行

1.7.3 クエリ

・ヘッダ部の最後の文字が ? マークであるメッセージユニットをクエリと呼びます。
設定内容や測定結果などを問い合わせるコマンドを意味します。本器はクエリを受信すると応答メッセージを作成し、出力キューに格納します。

・出力キューに応答メッセージ(の一部)が残っている状態で、新しいメッセージを受信すると、出力キューの内容はクリアされ、クエリエラー(サービス要求原因となる)になります。

・クエリの場合も、複数のメッセージユニットを 1 個のメッセージとして構成することが可能です。作成される応答メッセージも複数の応答メッセージユニットをセミコロン(;)でつないだものとなります。

・1 個のメッセージにまとめるクエリの組み合わせによってはクエリエラーとなることがあります。"第3部 機器固有コマンド"の項で使用するクエリについて詳細をご確認下さい。

1.7.4 ヘッダ部ニーモニック

・ヘッダ部のニーモニックを構成する文字は ASCII コードの A ～ Z と 0～9 及びアンダースコア(_) の組み合わせとなります。
・A ～ Z は大文字でも小文字でも両者の混在でも、同じ意味に解釈します。
・ほとんどのニーモニックには 3 文字あるいは 4 文字の省略形が存在します。
・本器がクエリに応じて作成する応答メッセージユニットの場合、
1)ヘッダ付きとするか または ヘッダ無しでデータ部だけで構成するか
2)ヘッダ付きの場合、ニーモニックをフルスペルとするか または 省略形とするか
をコマンドで選択することができます。
・本器が送信する応答メッセージのヘッダ部のニーモニックは必ず大文字となります。
・クエリによっては、必ずヘッダ付きとなるものや、逆にヘッダなしとなるものがあります。
"第3部 機器固有コマンド"の項で詳細をご確認下さい。

1.7.5 データ部フォーマット

a.＜文字プログラムデータ＞/＜文字応答データ＞
・＜文字プログラムデータ＞と＜文字応答データ＞はヘッダ部に使用されるニーモニックと同様に、A ～ Z と 0 ～ 9 及びアンダースコア(_) の組み合わせによって構成されるデータです。
:CHANNEL1:TRACE ON;:CHANNEL1:VDIV 50mV;:CHANNEL1:COUPLING DC
の例の場合、:CHANNEL1:TRACE コマンドのパラメータである ON と :CHANNEL1:COUPLING コマンドのパラメータである DC が＜文字プログラムデータ＞となります。
・ヘッダ部のニーモニックと同様に大文字/小文字の混在や省略形の使用が可能です。
・本器が＜応答メッセージ＞として作成する＜文字応答データ＞の場合、必ず大文字となります。

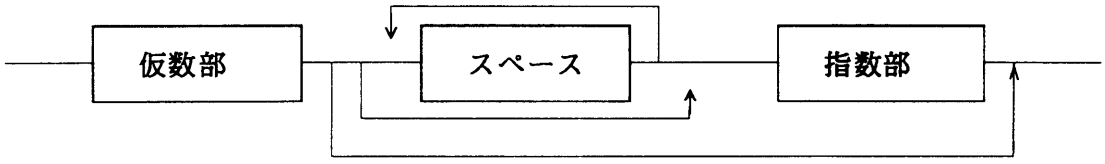
b.＜数値プログラムデータ＞/＜数値応答データ＞
・整数(NR1)･実数(NR2)･指数(NR3) のいずれかの形式で数値として扱われるデータに使用します。
:CHANNEL1:VDIV 50mV のメッセージの場合、50mV の部分が数値プログラムデータです。
以下の数値はすべて同じ値と解釈します。
0.05 = 50E-3 = 5e-2 = 5E-2V = 50mV
・V または mV のような単位などを表すサフィックス部分は V(ボルト)や Hz(ヘルツ)のように単位そのものを表すサフィックス単位部分と m(ミリ)やk(キロ)のように単位に対する倍率を表すサフィックス乗数部分とに分かれます。
・サフィックス単位としてどのような文字を許すかはコマンド毎に異なります。"第3部 機器固有コマンド"の項で、各コマンド毎にご確認下さい。
・本器が＜応答メッセージ＞として作成する数値応答データにはサフィックス部分はありません。
・サフィックス乗数を表すニーモニックは 表 1.7.1 のように定義されます。

表 1.7.1 サフィックス乗数を表すニーモニック

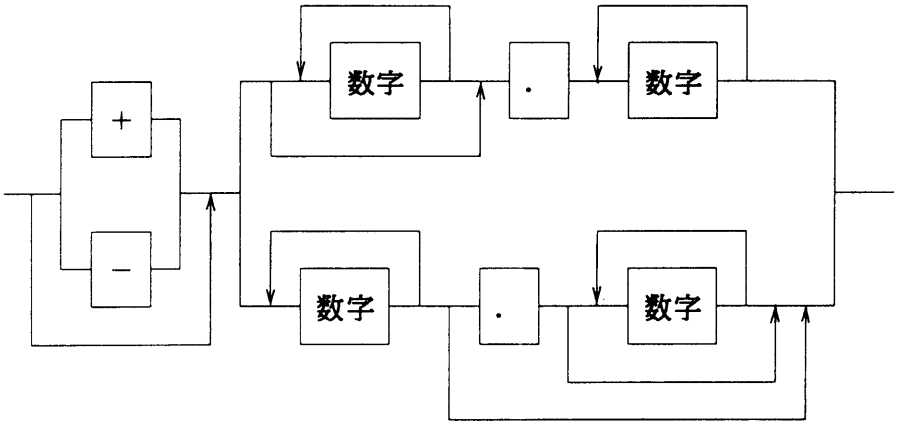
定 義	ニーモニック	定 義	ニーモニック
1E18	EX	1E-3	M
1E15	PE	1E-6	U
1E12	T	1E-9	N
1E9	G	1E-12	P
1E6	MA	1E-15	F
1E3	K	1E-18	A

ニーモニックは大文字/小文字の区別をしません。

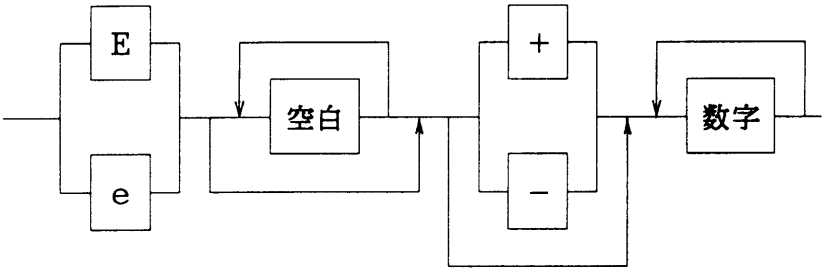
◇本器が受信する＜数値プログラムデータ＞のフォーマットは以下の通りです。
コマンドによっては、この後にサフィックス部を許します。



・仮数部のフォーマットを以下に示します。

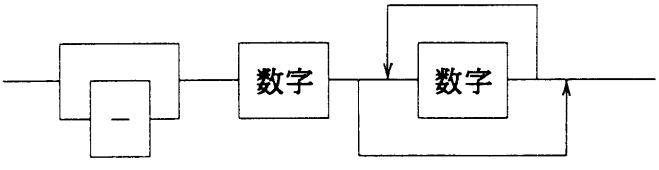


・指数部のフォーマットを以下に示します。

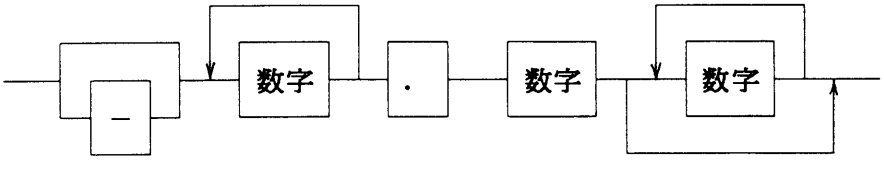


◇本器が送信する数値応答データのフォーマットは以下の通りです。

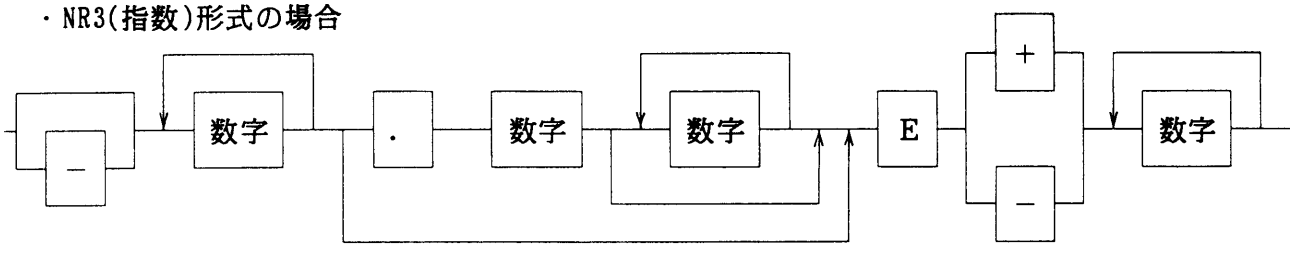
・NR1(整数)形式の場合



・NR2(実数)形式の場合



・NR3(指数)形式の場合



c.バイナリブロックデータ

・波形データなどの大量のデータを送受信する場合に使用します。
・本器がバイナリ転送を行う場合、コマンドによって以下の場合があります。
1)バイナリブロックを構成するデータ数が固定的に決まっているときと
2)バイナリブロックを構成するデータ数を設定できるとき
いずれの場合も、指定したデータ数と実際の転送データ数が違うと、その後の動作が異常になることがあります。特に、本器はバイナリブロックデータの転送に DMA 転送機能を使用しています。データ数の指定を誤ると GP-IB システムがハングアップすることがあります。IFC 信号と DCL または SDC コマンドにより、本器はハングアップ状態から復帰します。
・バイナリブロックデータ転送では、デリミタとして指定したコードとデータの区別がつきません。本器にバイナリデータを送る場合、デリミタなしでデータのみを転送してください。本器からバイナリデータを読み出す場合も、デリミタなしでデータだけが送られてきます。
・本器はバイナリブロックデータの最終バイトを送信するときに EOI 信号を真にします。
・本器がバイナリブロックデータを受信する場合は、指定したデータ数の受信を完了した場合と同様に、EOI 信号を受信してもバイナリブロックデータの最後と判断します。このとき、指定したデータ数と実際に受信したデータ数が違うとエラーとなります。
・本器はIEEE488.2に規定された＜任意ブロックプログラムデータ＞のうち、#0 で始まるパスには対応していませんので、ご注意下さい。

d.＜任意 ASCII 応答データ＞

・本器が作成する応答データで＜文字応答データ＞や数値応答データでは許されない ASCII 文字を含むようなテキストで構成されるデータです。
・＜任意 ASCII 応答データ＞を返すクエリとデリミタの間に新しいクエリメッセージユニットが存在した場合、それらのクエリに対する応答は作成せず、クエリエラーとします。

1.7.6 デリミタ(<PROGRAM/RESPONSE MESSAGE TERMINATOR>)

・本器が＜応答メッセージ＞を送信する場合のデリミタ(<RESPONSE MESSAGE TERMINATOR>)として、LF または CR LF が選択できます。
・EOI は必ず送信されます。
・デリミタの設定はシステム メニュー内 REMOTE I/F サブメニューで行います。"9.1 REMOTE I/F"の DELIMITER 設定をご参照下さい(リモート インタフェース経由の設定はできません)。
・応答データがバイナリデータの場合は、LF または CR LF は送信されず、EOI 信号が最終バイトに同期して送信されます。
・本器がプログラムメッセージを受信する場合は LF、CR LF または EOI のうちのいずれを受信してもデリミタ(<PROGRAM MESSAGE TERMINATOR>)として認識します。デリミタ設定には依存しないことにご注意下さい。

1.7.7 入出力バッファ

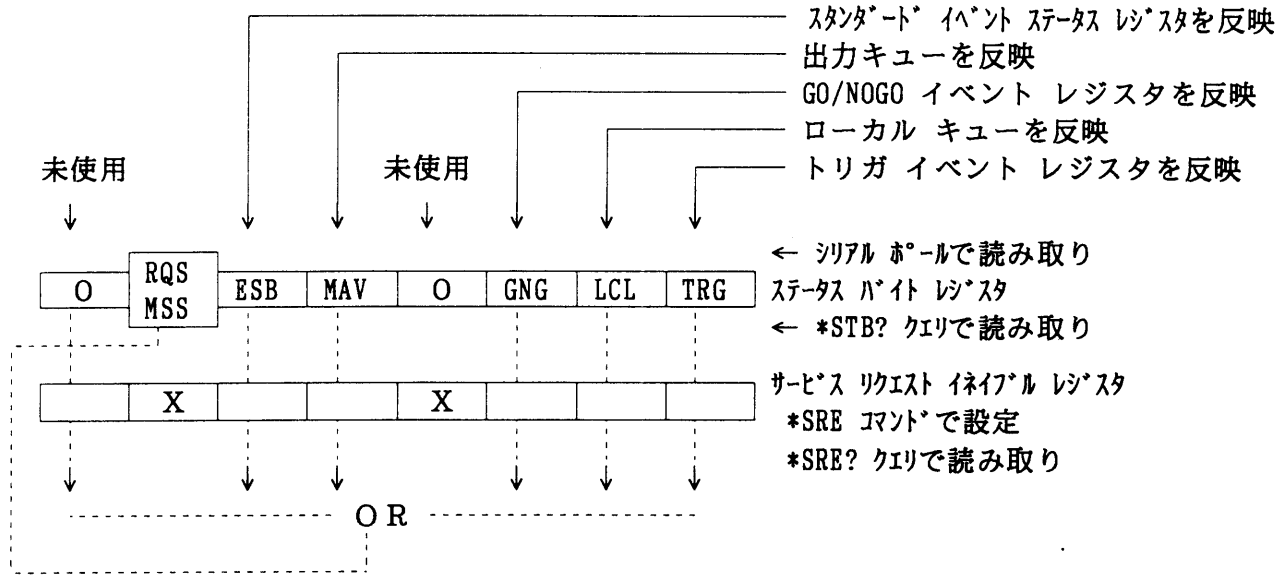
・本器は 512 バイトの入力バッファを持ちます。
・入力バッファがフルにならなくても、デリミタまで受信すると、バッファ内のコマンド解釈に入ります。コマンド解釈･実行中は次のメッセージの受信は行いません。新しいメッセージの先頭のバイトでハンドシェイクはホールドオフします。コマンド解釈･実行が終了するとハンドシェイクを再開します。
・入力バッファがフルになった場合、一旦ハンドシェイクをホールドオフし、バッファ内のメッセージユニットのコマンド解釈・実行を終了させてからハンドシェイクを再開します。
・本器は 512 バイトの出力バッファ(出力キュー)を持ちます。
・複数のクエリメッセージユニットを集めて 1 個のメッセージとした場合、応答メッセージの合計バイト数が 512 を越えないようにご注意下さい。
バイナリデータブロック転送の場合は、512 バイトを越えるデータ数でも問題ありません。

1.8 サービス要求とステータスレポート構造

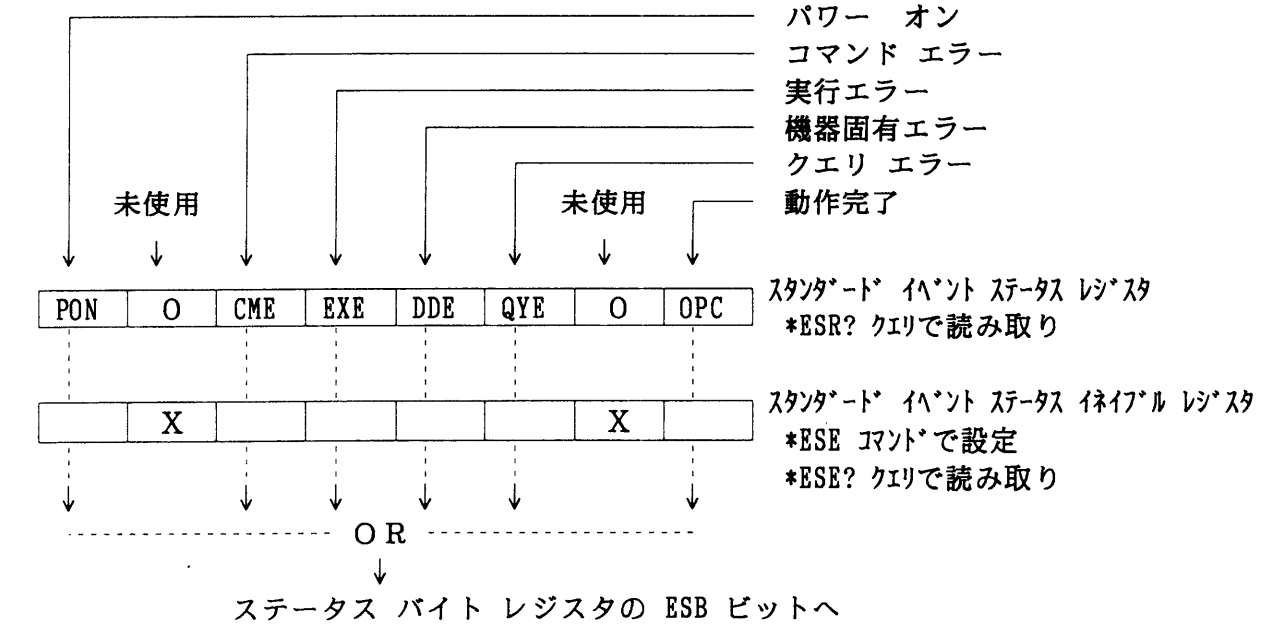
1.8.1 サービス要求(SRQ)

・本器は動作完了･各種エラー発生などによってサービス要求を出力(SRQラインを"L"に)します。
・サービス要求に使用されるバスラインがワイヤード OR 接続の SRQ ライン 1 本だけであるのに対して、本器はサービス要求発生要因を複数種類持っています。どの要因によってサービス要求が発せられたのかを知るためには、コントローラはシリアルポールを実行したり、本器の内部ステータスレジスタなどを読み取って判断する必要があります。

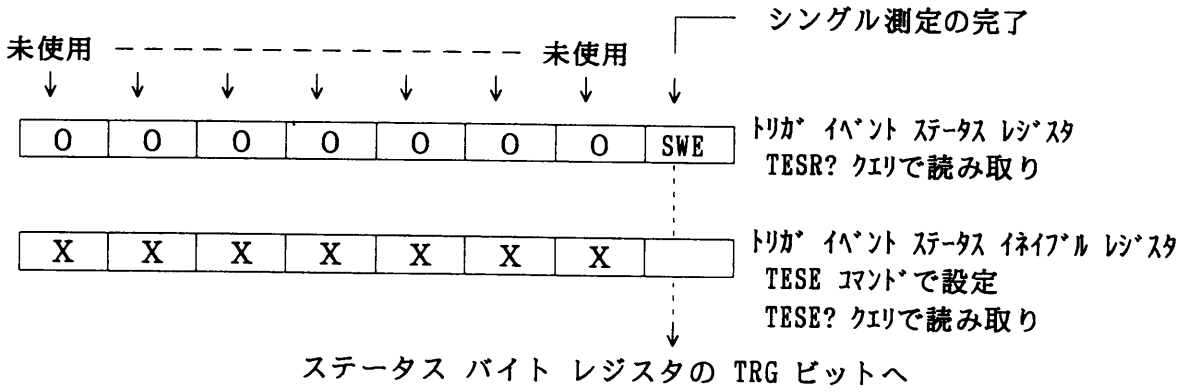
ステータスバイト



スタンダード イベント ステータス



トリガ イベント ステータス



1.8.2 ステータス バイト レジスタ

・GP-IB システムバス経由のシリアルポール動作 または *STB? クエリによる読み取りが可能です。
・階層構造をなす本器のステータス情報の最終段に位置します。
・レジスタは 8 ビット構成で、その内容は以下の通りです。

（ビット7） 本器はステータス バイト レジスタのビット 7 を使用しません。
常に 0 となります。

RQS(ビット6) シリアル ポールを実施するとステータス バイトのビット 6 として返されます。
ステータス バイト レジスタ内の他のビット(ESB,MAV,…,TRG)をサービス リクエスト イネイブル レジスタでマスクした結果の全ビットの論理 OR が 0→1 に変化したときにセットされます。電源投入時とシリアル ポール実施時にクリアされます。

MSS(ビット6) *STB? クエリ応答のビット 6 として返されます。
ステータス バイト レジスタ内の他のビット(ESB,MAV,…,TRG)をサービス リクエスト イネイブル レジスタでマスクした結果の全ビットの論理 OR を反映しています。

ESB(ビット5) シリアル ポール または *STB? クエリ応答のビット 5 として返されます。
スタンダード イベント ステータス レジスタに対応するサマリ メッセージです。

MAV(ビット4) シリアル ポール または *STB? クエリ応答のビット 4 として返されます。
出力キューに対応するサマリ メッセージです。

（ビット3） 本器はステータス バイト レジスタのビット 3 を使用しません。
常に 0 となります。

GNG(ビット2) シリアル ポール または *STB? クエリ応答のビット 2 として返されます。
GO/NOGO イベント ステータス レジスタ に対応するサマリ メッセージです。

LCL(ビット1) シリアル ポールあるいは *STB? クエリ応答のビット 1 として返されます。
ローカル キューに対応するサマリ メッセージです。

TRG(ビット0) シリアル ポールあるいは *STB? クエリ応答のビット 0 として返されます。
トリガ イベント ステータス レジスタに対応するサマリ メッセージです。

1.8.3 サービス リクエスト イネイブル レジスタ

・ステータス バイト レジスタをマスクするためのレジスタです。マスクした結果に応じて RQS や MSS ビットが設定されます。従って、サービス要求も影響されます。
・*SRE コマンドでマスク パターンの設定が *SRE? クエリで読み取りが可能です。
・レジスタの内容を電源投入時にクリア(=0)するかどうかは *PSC コマンドで設定可能です。
・デバイス クリアや *CLS コマンドには直接的には影響されません。

1.8.4 スタンダード イベント ステータス レジスタ

・IEEE488.2 規格で定められたレジスタです。
・*ESR? クエリ で読み出しが可能です。
・レジスタの内容をスタンダード イベント ステータス イネイブル レジスタでマスクした結果の全ビットの論理 OR がリアルタイムにステータス バイト レジスタの ESB ビットに反映します。
・レジスタの内容は *CLS コマンドと *ESR? による読み出しとの何れによっても、クリア(=0)します。
・レジスタは 8 ビット構成で、その内容は以下の通りです。

PON(ビット7) 電源投入時に 1 がセットされます。

(ビット6) 本器では使用しません。
常に 0 となります。

CME(ビット5) コマンド エラー発生時に 1 がセットされます。
コマンド シンタックスになんらかのエラーがあることを示します。

EXE(ビット4) 実行エラー発生時に 1 がセットされます。
コマンドの実行ができない または 正常に完了しなかったことを示します。

DDE(ビット3) 機器固有エラー発生時に 1 がセットされます。
本器固有のエラーを示します。

QYE(ビット2) クエリ エラー発生時に 1 がセットされます。
コントローラがクエリ コマンドを送らずに本器からメッセージを読もうとした または 応答メッセージを完全に読み取る前に次のメッセージを送ってきたなどのエラーを示します。

(ビット1) 本器では使用しません。
常に 0 となります。

OPC(ビット0) 動作完了時に1がセットされます。
*OPC コマンドによる、コントローラとデバイス間の同期機能をサポートします。

1.8.5 スタンダード イベント ステータス イネイブル レジスタ

・スタンダード イベント ステータス レジスタをマスクするためのレジスタです。
マスクした結果に応じてステータス バイト レジスタの ESB ビットが設定されます。
・*ESE コマンドでマスク パターンの設定が *ESE? クエリで読み出しが可能です。
・レジスタの内容を電源投入時にクリア(=0)するかどうかは *PSC コマンドで設定可能です。
・デバイス クリアや *CLS コマンドには影響されません。

1.8.6 出力キュー

・出力キューはコントローラへの応答メッセージを格納するための出力バッファです。
・各種のクエリに応じて適切な応答メッセージを出力キューに格納します。
・出力キューに 1 バイトでもデータがある限り、ステータス バイト レジスタの MAV ビットが 1 になります。
・出力キューの内容は電源投入時とデバイス クリア動作時にクリア(=空)します。
出力キューが空でない状態で、コントローラがすべてのデータ バイトを読み出す前に、新しいプログラム メッセージを送ってくるとクエリ エラーとなり、出力キューの内容もクリアされます。

1.8.7 GO/NOGO イベント ステータス レジスタ

・GO/NOGO 判定結果をレポートするためのイベント レジスタです。
・詳細は GO/NOGO(ｵﾌﾟｼｮﾝ)の取扱説明書をご参照ください。

1.8.8 GO/NOGO イベント ステータス イネイブル レジスタ

・GO/NOGO イベント ステータス レジスタをマスクするためのレジスタです。
マスクした結果に応じてステータス バイト レジスタの GNG ビットが設定されます。
・詳細は GO/NOGO(ｵﾌﾟｼｮﾝ)の取扱説明書をご参照ください。

1.8.9 トリガ イベント ステータス レジスタ

・シングル測定終了をレポートするためのイベント レジスタです。
・レジスタの内容をトリガ イベント ステータス イネイブル レジスタでマスクした結果の全ビットの論理 OR がリアルタイムにステータス バイト レジスタの TRG ビットに反映されます。
・レジスタは下位 1 ビットのみ使用します。それ以外のビットは常に 0 とみなします。
・レジスタに対応するクエリは :TESR? です。
・レジスタの内容は電源投入時にクリア(=0)します。

1.8.10 トリガ イベント ステータス イネイブル レジスタ

・トリガ イベント ステータス レジスタをマスクするためのレジスタです。
マスクした結果に応じて ステータス バイト レジスタの TRG ビットが設定されます。
・:TESE コマンドでマスク パターンの設定が :TESE? クエリで読み出しが可能です。
・レジスタの内容を電源投入時にクリア(=0)するかどうかは *PSC コマンドで設定可能です。
・デバイス クリアや *CLS コマンドには影響されません。

1.8.11 ローカル キュー

・REMOTE 状態から LOCAL 状態に戻ったときに、ローカルキューに 1 が書き込まれます。
・:LCLQ? クエリでローカル キューの内容が出力キューに転送されます。
・ローカル キューが空か否かがリアルタイムにステータス バイト レジスタの LCL ビットに反映されます。
・ローカル キューのサイズは 1 バイトです。
・ローカル キューの内容は電源投入時とデバイス クリア動作時と :LCLQ? クエリにより出力キューに転送された時と *CLS コマンドによりクリア(=空)します。

1.8.12 エラー キュー

・スタンダード イベント ステータス レジスタでレポートされる 4 種のエラーの詳細情報を格納するためのバッファです。
・エラー キューはエラー 10回分の情報を格納できます。FIFO 動作となります。
・:SYSTEM:ERROR? クエリで出力キューに 1回分の情報が転送されます。
・エラー キューが空か否かを直接に反映するステータス ビットなどはありません。
・エラー キューの内容は電源投入時と :SYSTEM:ERROR? クエリによる出力キューへ転送と読み取りを必要回数繰り返した時と *CLS コマンドによりクリア(=空)されます。

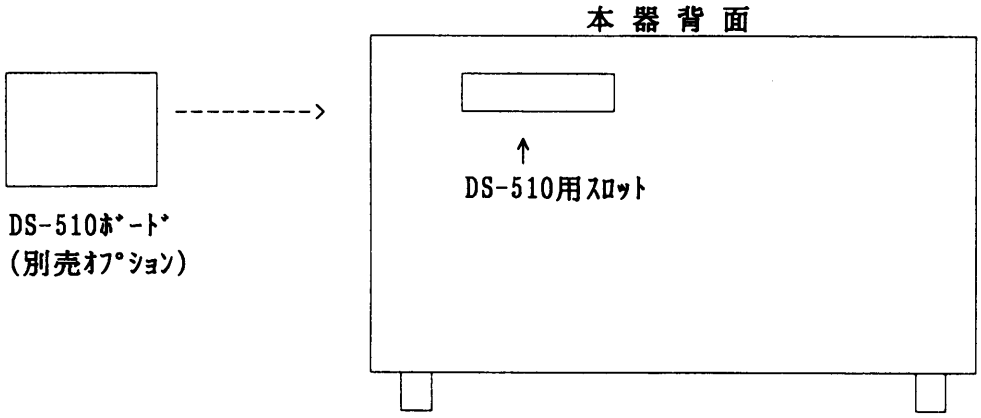
第２部　ＲＳ－２３２Ｃインタフェース

2.1 概　要

米国のEIA(アメリカ電子機会工業会:Electronic Industries Assosiation)が DTE(データ端末: Data Terminal Equipment)と DCE(回線終端装置:Data Circuit Terminal Equipment)とのインタフェース条件を決めた規格です。この規格ではモデム、パソコン周辺装置の入出力インタフェースとして広く使われています。

ユニットの取り付け

RS-232C インタフェースはオプションです。インタフェースユニット･DS-510 を背面の専用スロットに挿入します。



RS-232C 機能

DS-510 を本器に装着すると、以下の RS-232C 機能が追加されます。

・コントローラによるリモート制御
　パソコン等のコントローラと本器をケーブルで接続し、リモート状態で本器のパネル操作、データ送受信を行うことできます。

・プロッタへのハードコピー出力
　RS-232C インターフェース機能を備えた、HP-GL コマンド対応のプロッタに測定結果をハードコピー出力します。

2.2 仕　様

電気的、機械的仕様は EIA に準拠します。通信方式は半二重です。

2.3 設定条件

外部装置と本器との間でデータ通信を行うときは以下の設定をします。

・ボーレート
300,600,1200,2400,4800,9600 のいずれかを選択します。
設定方法は“取扱説明編の 9.1 リモートインタフェース(REMOTE I/F)”をご参照下さい。

・ビット長
ビット長は 8 ビット固定です。

・パリティビット
ODD(奇数)、EVEN(偶数)、NONE(なし)のいずれかを選択します。
設定方法は“取扱説明編の 9.1 リモートインタフェース(REMOTE I/F)”をご参照下さい。

・ストップビット
1 ビット または 2 ビットのいずれかを選択します。
設定方法は“取扱説明編の 9.1 リモートインタフェース(REMOTE I/F)”をご参照下さい。

・デリミタ
LF または CR/LF のいずれかを選択します。
設定方法は“取扱説明編の 9.1 リモートインタフェース(REMOTE I/F)”をご参照下さい。

2.4 信号ラインとコネクタピン

RS-232C の信号ラインとコネクタピンの関係を表2.4.1 に示します。

表2.4.1 信号ラインとコネクタピン

ピン番号	信　号	信号の方向 本器 ←→ 外部装置	機　　能
1	FG		匡体接地
2	SD	→	送信データ
3	RD	←	受信データ
4	RS	→	送信要求信号
5	CS	←	送信可能信号
6	NC		未使用
7	SG		信号用接地
8～25	NC		未使用

2.5 コネクタとケーブル

a.コネクタ
RS-232C で使用するコネクタは 25 ピン差し込み式です。一般に 25 ピン D-SUB コネクタと呼ばれています。

b.ケーブル
ケーブルの形状については規定していません。ケーブル長は最大 15m です。

c.外部機器との接続
電源を OFF にした後、本器と外部機器を接続してください。

2.6 コントローラによる制御

2.6.1 コマンド

本器を RS-232C インタフェースでリモート制御する場合、RS-232C 専用のコマンドを持ちます(表2.6.1 参照)。これ以外のコマンドは GP-IB のコマンドと共通です。"第3部 機器固有コマンド"をご参照ください。

注　　意

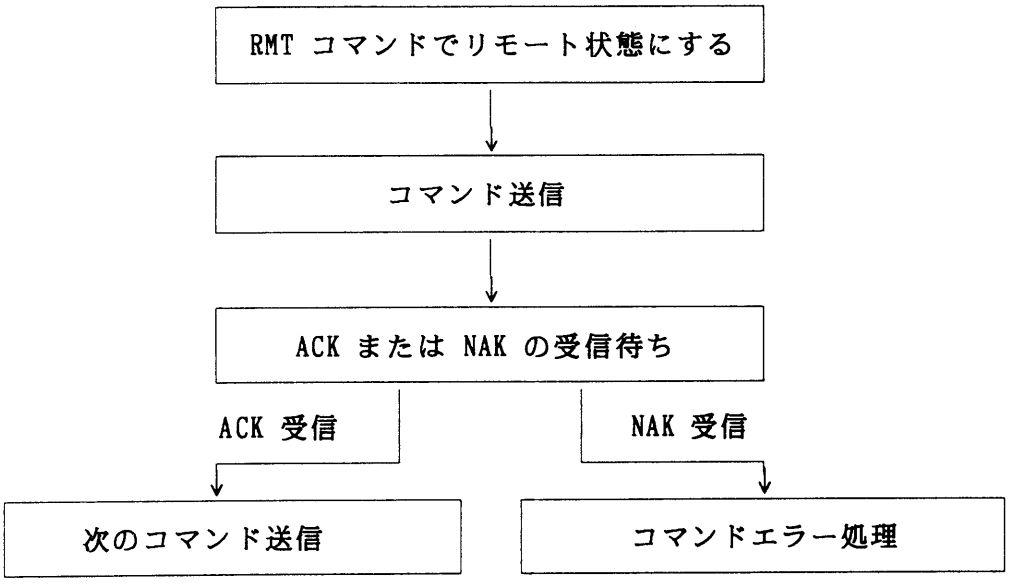
RS-232C 専用コマンドはマルチコマンドの一部にせず、そのコマンドだけで 1 個のプログラムになるようにしてください。

表2.6.1 RS-232C 専用コマンド

コマンド名	機　　能
RMT	本器を RS-232C のリモート状態に遷移させます。 コントローラはこのコマンドを一番最初に必ず出力します。
DCL	本器に対してデバイスクリアを行います。GP-IB と異なり、機器固有コマンドと同等のメッセージになりますので 測定、コピーなどの動作を強制停止させることはできません。

2.6.2 コントローラ側の処理

コントローラ側のリモートプログラムの作成に際しては以下の流れのように行って下さい。

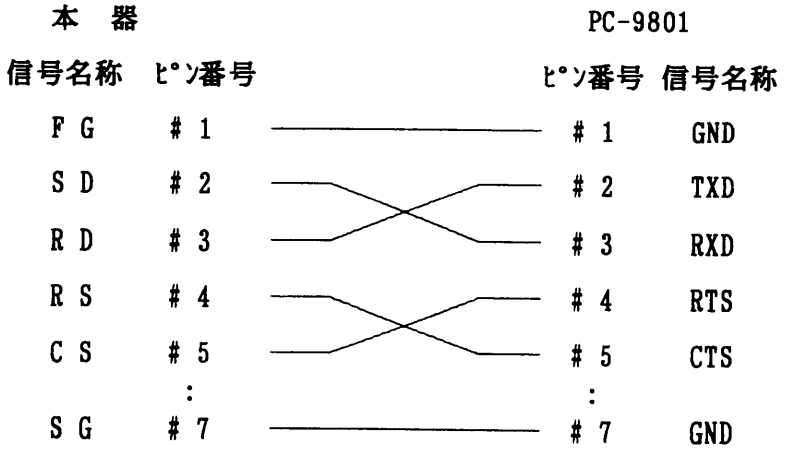


◇コマンド送信後にコントローラ側は必ず ACK(16 進で 06) または NAK(16 進で 15)を受信します。
- ACK を受信すれば次の処理へ進みます。
- NAK を受信した場合はコマンドエラーです。コマンドエラー処理を行って下さい。

2.7 PC-9801 との接続

PC-9801 との接続する場合の接続方法を図2.7.1 に示します。

図2.7.1 PC-9801 との接続



第３部　機器固有コマンド

機器固有コマンド

リモートコントロールに使用するコマンドについて説明します。

・本器は受信する＜プログラムメッセージ＞のヘッダ部や＜文字プログラムデータ＞のニーモニックは、大文字/小文字いずれでも正規のメッセージとして認識します。また、本器が＜応答メッセージ＞として送信する場合には、これらのニーモニックを必ず大文字とします。
・リモートコマンドの説明で、大文字と小文字が組み合わされたニーモニックとなっている場合には、小文字の部分は省略形を使用する場合に省略される部分であることを示します。
[例]:TRIGger:SWEep と表記されたコマンド
TRIGGER 親コマンドの正式表記は TRIGGER で省略形が TRIG です。
SWEEP サブコマンドの正式表記は SWEEP で省略形が SWE です。
・本器が受信する＜プログラムメッセージ＞のヘッダ部としては、以下のいずれも正規のコマンドとして認識され、エラーになりません。
:TRIGGER:SWEEP
:TRIG:SWE
:trigger:sweep
:trig:swe
:Trigger:Swe
:TRIG:sweep
・本器が＜応答メッセージ＞として送信する場合のヘッダ部は以下のいずれかとなります。
:TRIGGER:SWEEP　　(:SYSTem:LONGform 設定が ON の場合)
:TRIG:SWE　　(:SYSTem:LONGform 設定が OFF の場合)
・コマンド一覧表でニーモニックを区切っている | 記号は「または」を意味します。
[例]:TRIGger:SWEep ALEVel|AUTO|NORMal|SINGle と表記されている場合
:TRIGger:SWEep コマンドのパラメータとして ALEVel、AUTO、NORMal または SINGle のいずれかのニーモニックを使用すべきであることを示します。
・コマンドリストを表3.1～表3.5 に示します。詳細は 3-8 ﾍﾟｰｼﾞ以降をご参照ください。

コマンドリスト

表3.1 コマンドリスト(1/5)

	コマンド	クエリ	備　　考	参　照
共　　通	——	*IDN?	本器の識別	3.1.1
	——	*OPT?	本器の装備しているオプションの識別	3.1.2
	——	*LRN?	セットアップデータの一括読み出し	3.1.3
	*RST	——	セットアップ条件の初期化	3.1.4
	——	*TST?	RAM、ROM テスト	3.1.5
	*OPC	*OPC?	動作完了メッセージによるアプリケーションとの同期	3.1.6
	*WAI	——	動作終了待ちによるｱﾌﾟﾘｹｰｼｮﾝとの同期	3.1.7
	*CLS	——	クリアステータス	3.1.8
	*ESE	*ESE?	スタンダードイベントステータスイネイブルレジスタの設定と読み出し	3.1.9
	——	*ESR?	スタンダードイベントステータスイネイブルレジスタの読み出しとクリア	3.1.10
	*PSC	*PSC?	POWER ON 時のイネーブルレジスタのクリアと読み出し	3.1.11
	*SRE	*SRE?	サービスリクエストイネイブルレジスタの設定と読み出し	3.1.12
	——	*STB?	ステータスバイトの読み出し	3.1.13
	*TRG	——	ラン(RUN)にする	3.1.14
	*RCL	——	セットアップ条件のリコール	3.1.15
	*SAV	——	セットアップ条件のセーブ	3.1.16
ルートレベル	RUN	——	ラン(RUN)にする	3.2.1
	STOP	——	ストップ(STOP)にする	3.2.2
	CLEer	——	パーシスタンス表示をクリア	3.2.3
	ASCHannel	ASCHannel?	ｵｰﾄｾｯﾄｱｯﾌﾟﾓｰﾄﾞの設定と読み出し	3.2.4
	ASETup	——	オートセットアップの実行	3.2.5
	WAIT	——	SINGle の場合、測定完了待に設定	3.2.6
	WSGL	——	SINGle にし、ラン(RUN)にする	3.2.7
	——	TESR?	トリガイベントステータスレジスタの読み出しとクリア	3.2.8
	TESE	TESE?	トリガイベントステータスイネイブルレジスタの設定と読み出し	3.2.9
	——	LCLQ?	ローカルキューの読み出し	3.2.10
ACQuire 関連	MEMory	MEMory?	メモリ使用種別の設定と読み出し	3.3.1
	POINts	POINts?	メモリ長の設定と読み出し	3.3.2
	TYPE	TYPE?	波形処理モードの設定と読み出し	3.3.3
	COUNts	COUNts?	ｱﾍﾞﾚｰｼﾞ、ﾏｯｸｽﾎｰﾙﾄﾞ回数設定と読み出し	3.3.4
	INTerpolate	INTerpolate?	補間方式の設定と読み出し	3.3.5
	RPAGe	RPAGe?	ランページの設定と読み出し	3.3.6
	DPAGe	RPAGe?	表示ページの設定と読み出し	3.3.7

表3.2 コマンドリスト(2/5)

	コマンド	クエリ	備　　考	参　照
TIMebase関連	SAMPling	SAMPling?	サンプリングモードの設定と読み出し	3.4.1
	SDIV	SDIV?	SEC/DIV の設定と読み出し	3.4.2
	SCRoll	SCRoll?	スクロール時間の設定と読み出し	3.4.3
	BSWeep	BSWeep?	B 掃引の ON/OFF の設定と読み出し	3.4.4
	DISPlay	DISPlay?	B 掃引ディスプレイの設定と読み出し	3.4.5
	BSDiv	BSDiv?	B 掃引時間の設定と読み出し	3.4.6
	DELay	DELay?	遅延時間の設定と読み出し	3.4.7
	TSEParation	TSEParation?	ﾄﾚｰｽｾﾊﾟﾚｰｼｮﾝの設定と読み出し	3.4.8
CHannel 関連	VDIV	VDIV?	ﾁｬﾈﾙn の垂直感度の設定と読み出し	3.5.1
	ZOOM	ZOOM?	ﾁｬﾈﾙn のズームの設定と読み出し	3.5.2
	COUPling	COUPling?	ﾁｬﾈﾙn の入力結合の設定と読み出し	3.5.3
	POSition	POSition?	ﾁｬﾈﾙn の垂直位置の設定と読み出し	3.5.4
	TRACe	TRACe?	ﾁｬﾈﾙn のトレース表示の設定と読み出し	3.5.5
	INVert	INVert?	ﾁｬﾈﾙn のインバートの設定と読み出し	3.5.6
	FILTer	FILTer?	ﾁｬﾈﾙn のフィルタの設定と読み出し	3.5.7
	PROBe:ATT	PROBe:ATT?	ﾁｬﾈﾙn のﾌﾟﾛｰﾌﾞの減衰比設定と読み出し	3.5.8
	PROBe:AUTO	PROBe:AUTO?	ﾁｬﾈﾙn のﾌﾟﾛｰﾌﾞの AUTO 設定と読み出し	3.5.8
TRIGger 関連	MODE	MODE?	トリガモードの設定と読み出し	3.6.1
	SWEep	SWEep?	掃引方式の設定と読み出し	3.6.2
	SOURce	SOURce?	トリガ源の設定と読み出し	3.6.3
	PATH	PATH?	サブコマンドの対象となるトリガ源の設定と読み出し	3.6.4
	LEVel	LEVel?	トリガレベルの設定と読み出し	3.6.5
	SLOPe	SLOPe?	トリガスロープの設定と読み出し	3.6.6
	COUPing	COUPing?	トリガ結合の設定と読み出し	3.6.7
	REJect	REJect?	リジェクションの設定と読み出し	3.6.8
	NREJect	NREJect?	ノイズリジェクションの設定と読み出し	3.6.9
	POSition	POSition?	トリガポジションの設定と読み出し	3.6.10
	HOLDoff	HOLDoff?	ホールドオフ時間の設定と読み出し	3.6.11
	EMODe	EMODe?	イベントモードの設定と読み出し	3.6.12
	ECOunts	ECOunts?	イベントカウンタの設定と読み出し	3.6.13
	TIMer	TIMer?	イベントバリューの設定と読み出し	3.6.13
	CCHannel	CCHannel?	クロックチャンネルの設定と読み出し	3.6.14
	WHEN	WHEN?	WHEN の設定と読み出し	3.6.14
	LSTate	LSTate?	ステートの設定と読み出し	3.6.14
	TVSelect	TVSelect?	テレビセレクトの設定と読み出し	3.6.15
	TVLine	TVLine?	テレビラインの設定と読み出し	3.6.15
	TVMode	TVMode?	テレビモードの設定と読み出し	3.6.15

表3.3 コマンドリスト(3/5)

	コマンド	クエリ	備　　考	参　照
ARM 関連	SOURce	SOURce?	ｱｰﾑｿｰｽﾁｬﾈﾙの設定と読み出し	3.7.1
	LEVel	LEVel?	ｱｰﾑﾄﾘｶﾞﾚﾍﾞﾙの設定と読み出し	3.7.2
	SLOPe	SLOPe?	ｱｰﾑﾄﾘｶﾞｽﾛｰﾌﾟの設定と読み出し	3.7.3
	COUPling	COUPling?	ｱｰﾑﾄﾘｶﾞｶｯﾌﾟﾘﾝｸﾞの設定と読み出し	3.7.4
	REJect	REJect?	ｱｰﾑﾘｼﾞｪｸｼｮﾝの設定と読み出し	3.7.5
	NREJect	NREJect?	ｱｰﾑﾉｲｽﾞﾘｼﾞｪｸｼｮﾝの設定と読み出し	3.7.6
DISPlay 関連	SPLit	SPLit?	スプリット表示の設定と読み出し	3.8.1
	GRATicule	GRATicule?	グラテキュールの設定と読み出し	3.8.2
	MODE	MODE?	ディスプレイモードの設定と読み出し	3.8.3
	PERSistence	PERSistence?	パーシスタンス回数の設定と読み出し	3.8.4
	TEXT:DATA	TEXT:DATA?	コメント文字列の設定と読み出し	3.8.5
	TEXT:CLEAR	——	コメント文字列のクリア	3.8.5
	TEXT:STATe	TEXT:STATe?	コメント文字表示の設定と読み出し	3.8.5
	XY<x>	XY<x>?	X-Y 表示の設定と読み出し	3.8.6
	CALC:FUNC	CALC:FUNC?	演算子と対象波形の設定と読み出し	3.8.7
	CALC:VPOS	CALC:VPOS?	演算波形の垂直位置の設定と読み出し	3.8.7
	DATE	DATE?	日付け表示の設定と読み出し	3.8.8
	RGB	RGB?	RGB 出力の設定と読み出し	3.8.9
REFerence 関　連	DISPlay	DISPlay?	ﾘﾌｧﾚﾝｽ波形表示の設定と読み出し	3.9.1
	XPOSition	XPOSition?	水平軸位置の設定と読み出し	3.9.2
	YPOSition	YPOSition?	垂直軸位置の設定と読み出し	3.9.3
	SET	——	リファレンス波形の設定	3.9.4
	DATA	DATA?	ﾘﾌｧﾚﾝｽ波形データの設定と読み出し	3.9.5
CURsor 関 連	TMOVe	TMOVe?	Δtカーソルモードの設定と読み出し	3.10.1
	TPReference	TPReference?	Δtﾘﾌｧﾚﾝｽｶｰｿﾙ位置の設定と読み出し	3.10.2
	TPRset	——	Δtﾘﾌｧﾚﾝｽｶｰｿﾙｾｯﾄの実行	3.10.3
	TPMeasure	TPMeasure?	Δtﾒｼﾞｬﾒﾝﾄｶｰｿﾙ位置の設定と読み出し	3.10.4
	TFUNction	TFUNction?	Δt 機能の設定と読み出し	3.10.5
	TSOurce	TSOurce?	Δt 対象時間軸の設定と読み出し	3.10.6
	TRENtry	——	Δt レシオの基準設定	3.10.7
	——	TDELta?	Δt カーソル測定結果の読み出し	3.10.8
	VPRefernce	VPRefernce?	ΔVﾘﾌｧﾚﾝｽｶｰｿﾙ位置の設定と読み出し	3.10.9
	VPMeasure	VPMeasure?	ΔVﾒｼﾞｬﾒﾝﾄｶｰｿﾙ位置の設定と読み出し	3.10.10
	VFUNction	VFUNction?	ΔV 機能の設定と読み出し	3.10.11
	VSOurece	VSOurece?	ΔV 対象時間軸の設定と読み出し	3.10.12
	VRENtry	——	ΔV レシオの基準設定	3.10.13
	——	VDELta?	ΔV カーソル測定結果の読み出し	3.10.14

表3.4 コマンドリスト (4/5)

	コマンド	クエリ	備　　考	参　照
COPY 関　連	DEVice	DEVice?	コピーデバイスの設定と読み出し	3.11.1
	ROLL	ROLL?	ロールプリントの設定と読み出し	3.11.2
	FEED	——	紙送りの実行	3.11.3
	EXECute	——	コピーの実行	3.11.4
PLOT 関連	P1	P1?	P1 位置の設定と読み出し	3.12.1
	P2	P2?	P2 位置の設定と読み出し	3.12.2
	ROTation	ROTation?	ローテション位置の設定と読み出し	3.12.3
	PORT	PORT?	インタフェースの設定と読み出し	3.12.4
	CURSor	CURSor?	カーソルプロットの設定と読み出し	3.12.5
	READout	READout?	ﾘｰﾄﾞｱｳﾄﾌﾟﾛｯﾄの設定と読み出し	3.12.6
	GRATicule	GRATicule?	ｸﾞﾗﾃｷｭｰﾙﾌﾟﾛｯﾄの設定と読み出し	3.12.7
	PEN	PEN?	ペン番号の設定と読み出し	3.12.8
	DUMMy	——	ダミープロットの実行	3.12.9
SYSTEM 関 連	DATE	DATE?	日付けの設定と読み出し	3.13.1
	TIME	TIME?	時刻の設定と読み出し	3.13.2
	HEADer	HEADer?	ヘッダー部の設定と読み出し	3.13.3
	LONGform	LONGform?	ヘッダー部形式の設定と読み出し	3.13.4
	AVERage	AVERage?	アベレージモードの設定と読み出し	3.13.5
	——	ERRor?	エラーステータスの読み出し	3.13.6
	SET	SET?	ｾｯﾄｱｯﾌﾟﾃﾞｰﾀの一括設定と読み出し	3.13 7
	BALance	——	BALANCE 校正の実行	3.13.8
	GAIN	——	GAIN 校正の実行	3.13.9
	ADC	——	ADC 校正の実行	3.13.10
	TRIG	——	TRIG 校正の実行	3.13.11
	SPEed	SPEed?	リモートコントロールスピードの設定と読み出し	3.13.12
SETUP 関 連	SAVE	——	セットアップ条件のセーブ	3.14.1
	RECall	——	セットアップ条件のリコール	3.14.2
	DELete	——	ファイルの削除	3.14.3
	——	DIRectry?	ディレクトリイ情報の読み出し	3.14.4
	INITialize	——	すべてのファイルの削除	3.14.5
	SECurity	SECurity?	セキュリティの設定と読み出し	3.14.6

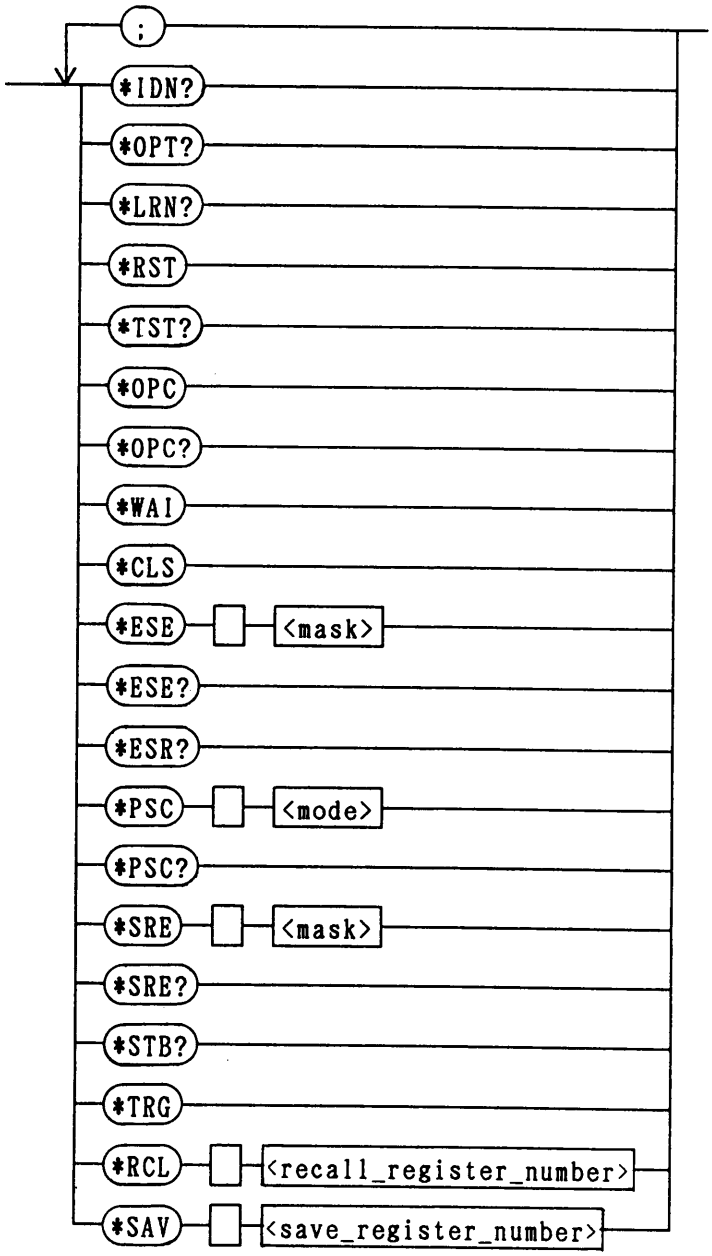
表3.5 コマンドリスト (5/5)

	コマンド	クエリ	備　　考	参　照
MCARd 関連	CDIRectry	CDIRectry?	対象データの設定と読み出し	3.15.1
	SAVE	——	メモリカードにセーブ	3.15.2
	RECall	——	メモリカードからリコール	3.15.3
	DELete	——	メモリカードのファイルの削除	3.15.4
	——	DIRectry?	ﾌｧｲﾙのディレクトリイ情報の読み出し	3.15.5
	INITialize	——	メモリカードの初期化	3.15.6
	SOURce	SOURce?	ﾒﾓﾘｶｰﾄﾞにｿｰｽﾁｬﾈﾙの設定と読み出し	3.15.7
	DESTination	DESTination?	ﾒﾓﾘｶｰﾄﾞの対象ﾄﾚｰｽの設定と読み出し	3.15.8
	——	READ?	ﾒﾓﾘｶｰﾄﾞからファイルを読み出し	3.15.9
	WRITe	——	メモリカードに書き込み	3.15.10
	SECurity	SECurity?	セキュリティの設定と読み出し	3.15.11
WAVeform関連	FORMat	FORMat?	転送フォーマットの設定と読み出し	3.16.1
	BORDer	BORDer?	バイト順序の設定と読み出し	3.16.2
	SOURce	SOURce?	波形転送対象チャネルのワード数の設定と読み出し	3.16.3
	POINts	POINts?	波形ﾃﾞｰﾀのワード数の設定と読み出し	3.16.4
	STARt	STARt?	データ転送開始番地の設定と読み出し	3.16.5
	——	DATA?	波形データの読み出し	3.16.6
	——	TYPE?	データタイプの読み出し	3.16.7
	——	XSIZe?	総波形データ長の読み出し	3.16.8
	——	XSCale?	時間軸方向の分解能の読み出し	3.16.9
	——	YSCale?	電圧軸方向の分解能の読み出し	3.16.10
	——	XOFFset?	トリガ点から波形データのスタート点迄の時間の読み出し	3.16.11
	——	YOFFset?	波形データ値 0 に対応する絶対値電圧の読み出し	3.16.12

3.1 共通コマンド

IEEE Std.488.2 規格で規定された共通コマンドうち、本器が備えるコマンドの一覧を次に示します。

図3.1.1 共通コマンド全体像



3.1.1 *IDN? クエリ(Identification Number)

本クエリでシステムインタフェースを介して本器の識別を行います。

クエリシンタックス　*IDN?
　　ニーモニック　*IDN?
　　省略形　*IDN?（省略形なし）

応答メッセージ　IWATSU,DS-9121 または 9122,<serial_number>,<software_revision><NL>
:SYSTem:HEADer 設定 ON/OFF に関わりなく、ヘッダ部はありません。
応答はコンマで区切られた 4 個のフィールドにより構成され、各フィールドは以下の通りです。

フィールド 1	メーカー名	IWATSU
フィールド 2	モデル名	DS-9121 または 9122
フィールド 3	シリアルナンバー	ASCII数値(<NR1>)
フィールド 4	ファームウェアレベル	ASCII数値(<NR2>)

エラー処理
- 応答データは、4 個のフィールド全体で＜任意 ASCII 応答データ＞フォーマットを構成するため、*IDN? クエリは＜プログラムメッセージ＞中の最後のクエリでなくてはなりません。これに違反する場合、クエリエラーになり *IDN? 以降のクエリに対する応答メッセージの作成、送信を行いません。
- クエリエラーが発生した場合、エラーキューに以下のメッセージを格納します。
  -440,"Query UNTERMINATED after indefinite response"

例　PRINT@DSO;"*IDN?"

3.1.2 *OPT? クエリ(Option)

本クエリでシステムインタフェースを介して本器の装備しているオプションを識別します。

クエリシンタックス　*OPT?
　　ニーモニック　*OPT?
　　省略形　*OPT?(省略形なし)

応答メッセージ　<option_number>,<option_number>,……,<option_number><NL>
・:SYSTem:HEADer設定 ON/OFF に関わりなく、ヘッダ部はありません。
・装着しているオプションの型番を示す文字列を出力します。複数のオプションを同時に装着している場合は、各オプションを示す文字列フィールドをカンマで区切って出力します。
・各フィールドは以下のルールに従います。
1)各フィールドの内容は DS- の 3 文字とそれに続く 3 桁の数字(合計 6 文字)からなるオプション型番となります。
例えば、内蔵プリンタのみを装着している場合には DS-900 の 6 文字が応答データとして返送されます。
2)オプションがない場合には ASCII 文字の数字 0 を応答データとします。

エラー処理
・応答データは 4 個のフィールド全体で<任意 ASCII 応答データ>フォーマットを構成するため、*IDN? クエリは<プログラムメッセージ>中の最後のクエリでなくてはなりません。これに違反する場合、クエリエラーになり、*IDN? 以降のクエリに対する応答メッセージの作成、送信を行いません。
・クエリエラーが発生した場合、エラーキューに以下のメッセージを格納します。
-440,"Query UNTERMINATED after indefinite response"

例　PRINT@DSO;"*OPT?"

3.1.3 *LRN? クエリ(Learn)

本クエリで その時点の本器のセットアップデータの一括読み出しをします。

クエリシンタックス　*LRN?
　　ニーモニック　*LRN?
　　省略形　*LRN?（省略形なし）

応答メッセージ　:SYSTem:SET #8<block_length_byte><binary_block>
- #8 は応答メッセージが＜決定長任意ブロック応答データ＞フォーマットであり、この後に続く<block_length_byte>が 8 桁の ASCII 数字列であることを示します。
- <block_length_byte> は 8 桁の ASCII 数字列で、この後に続く<binary_block> のバイト数を示します。リーディングゼロをサプレスせずに、必ず 8 桁の符号なし整数とします。
- <binary_block> は本器のセットアップデータをバイナリ値のブロックデータとしたものです。
- 後に、この応答データをコントローラが :SYSTem:SET コマンドを使用して送り返してきた場合に、本器は *LRN? クエリを受信したときと同じセットアップ状態に戻ります。

例　PRINT@DSO;"*LRN?"

3.1.4 *RST コマンド(Reset)

本コマンドで内部のセットアップを工場出荷時の規定値に初期化します。即ち、SET000 ファイルの設定をリコールします。

コマンドシンタックス　*RST
　　ニーモニック　*RST
　　省略形　*RST(省略形なし)

例　PRINT@DSO;"*RST"

3.1.5 *TST? クエリ(Test)

本クエリで内部の ROM テストと RAM テストを行い、エラー無しでテストを完了したか否かを示す応答を出力キューに格納します。

クエリシンタックス　*TST?
　　ニーモニック　*TST?
　　省略形　*TST?(省略形なし)

応答メッセージ　<result>
- :SYSTem:HEADer 設定 ON/OFF に関わりなく、ヘッダ部はありません。
- <result> は＜NR1 数値応答データ＞フォーマットでテスト結果を示します。
  パスの場合　:　0
  テストエラー:-330

例　PRINT@DSO;"*TST?"

3.1.6 *OPC コマンド/ *OPC? クエリ(Operation Complete)

*OPC コマンド

本コマンドを受信すると、このコマンドに先行するコマンド または クエリにより指定されたデバイス動作が総て完了した時点で、スタンダードイベントステータスレジスタの動作完了メッセージ(OPC ビット)を 1 にセットします。

コマンドシンタックス *OPC
ニーモニック *OPC
省略形 *OPC(省略形なし)

例 PRINT@DSO;"*OPC"

*OPC? クエリ

本クエリは OPC ビットをセットする代わりに ASCII 文字 1 を出力キューに格納します。スタンダードイベントステータスレジスタにセットしません。

クエリ シンタックス *OPC?
ニーモニック *OPC?
省略形 *OPC?(省略形なし)

応答メッセージ 1
応答データは<NR1 数値応答データ>フォーマットの 1 です。
:SYSTem:HEADer 設定 ON/OFF に関わりなく、ヘッダ部はありません。

例 PRINT@DSO;"*OPC?"

3.1.7 *WAI コマンド(Wait)

本コマンドを受信すると、このコマンドに先行するコマンド または クエリにより指定されたデバイス動作が総て完了するまで、以後のコマンドやクエリの実行を保留します。
本器の掃引方式をシングルにした場合、アプリケーションプログラムと本器の波形データのアクイジション完了との同期をとるために *WAI コマンドを使用します。

コマンドシンタックス *WAI
ニーモニック *WAI
省略形 *WAI(省略形なし)

例 PRINT@DSO;"*WAI"

3.1.8 *CLS コマンド(Clear Status)

本コマンドで本器のスタンダードイベントステータスレジスタ、トリガイベントレジスタ、GO/NOGO イベントレジスタ(ｵﾌﾟｼｮﾝ) および ローカルキューとエラーキューをクリアします。さらに、これらのレジスタまたはキューの内容を反映しているステータスバイトレジスタ内のビットをクリアします。

コマンドシンタックス *CLS
ニーモニック *CLS
省略形 *CLS(省略形なし)

例 PRINT@DSO;"*CLS"

3.1.9 *ESE コマンド/ *ESE? クエリ(Event Status Enable)

***ESE コマンド**

本コマンドでスタンダードイベントステータスイネイブルレジスタの各ビットを設定します。

コマンドシンタックス　*ESE <mask_arg>
　　ニーモニック　*ESE
　　省略形　*ESE(省略形なし)

・<mask_arg> はスタンダードイベントステータスイネイブルレジスタ設定用のパラメータで、＜数値プログラムデータ＞フォーマットの 0～255 の範囲の数値です。
・スタンダードイベントステータスレジスタは各ビットに 1 つづつ意味が割り振られており、イネイブルレジスタの対応するビットを 1 にしたときに対応する事象がイネイブル(ステータスバイトレジスタに反映される)、0 にするとディスエイブル(ステータスバイトレジスタに反映されない)となります。

例　PRINT@DSO;"*ESE 255"

***ESE? クエリ**

本クエリでスタンダードイベントステータスイネイブルレジスタの現在の内容を読み出します。

クエリシンタックス　*ESE?
　　ニーモニック　*ESE?
　　省略形　*ESE?(省略形なし)

応答メッセージ　<mask>

・:SYSTem:HEADer ON/OFF 設定に関わりなく、ヘッダ部はありません。
・<mask> はスタンダードイベントステータスイネイブルレジスタの各ビットの設定を 2 のべき乗で重み付けした数値で、＜NR1 数値応答データ＞フォーマットの 0～255 の範囲の数値です。

例　PRINT@DSO;"*ESE?"

備　考　スタンダードイベントステータスイネイブルレジスタ構成を表3.1.1 に示します。

表3.1.1 スタンダードイベントステータスイネイブルレジスタ構成

ビット	重み付け	意　味
7	128	PON- パワーオン
6	64	URQ- 本器では使用しない
5	32	CME- コマンドエラー
4	16	EXE- 実行エラー
3	8	DDE- 機器固有エラー
2	4	QYE- クエリエラー
1	2	RQC- 本器では使用しない
0	1	OPC- 動作完了

3.1.10 *ESR? クエリ(Event Status Register)

本クエリでスタンダードイベントステータスレジスタの現在の内容を読み出します。
読み出した後、スタンダードイベントステータスレジスタの内容をクリアします。

クエリシンタックス　*ESR?
　　ニーモニック　*ESR?
　　省略形　*ESR?(省略形なし)

応答メッセージ　<status>
・:SYSTem:HEADer ON/OFF 設定に関わりなく、ヘッダ部はありません。
・<status> はスタンダードイベントステータスレジスタの各ビットの値を 2 のべき乗で重み付けした数値で <NR1 数値応答データ> フォーマットの 0～255 の範囲の数値です。

例　PRINT@DSO;"*ESR?"

備　考　スタンダードイベントステータスレジスタ構成を表3.1.2 に示します。

表3.1.2 スタンダードイベントステータスレジスタ構成

ビット	重み付け	意　味
7	128	PON- パワーオン
6	64	URQ- 本器では使用しないため、常に0
5	32	CME- コマンドエラー
4	16	EXE- 実行エラー
3	8	DDE- 機器固有エラー
2	4	QYE- クエリエラー
1	2	RQC- 本器では使用しないため、常に0
0	1	OPC- 動作完了

3.1.11 *PSC コマンド/ *PSC? クエリ(Power on Status Clear)

*PSC コマンド

サービスリクエストイネイブルレジスタとスタンダードイベントステータスイネイブルレジスタとトリガイベントイネイブルレジスタと GO/NOGO イベントイネイブルレジスタ(ｵﾌﾟｼｮﾝ)の電源投入時の自動クリアをコントロールします。

コマンドシンタックス *PSC <psc_flag_arg>
ニーモニック *PSC
省略形 *PSC(省略形なし)

・<psc_flag_arg> はパワーオンステータスクリアフラグ設定用のパラメータで、-32767～+32767 の範囲の＜数値プログラムデータ＞フォーマットの数値です。
・0 に設定すると、各イネイブルレジスタの状態が電源断の間も保持されます。
・1 (0以外)に設定すると、各イネイブルレジスタは電源再投入時にクリアされます。

例 PRINT@DSO;"*PSC 0"

*PSC? クエリ

本クエリで設定値を知ることができます。応答データの値が 0 であれば、各イネイブルレジスタの状態が電源断の間も保持されることを示しています。応答データの値が 1 の場合、上記各イネイブルレジスタは電源再投入時にクリアされることを示しています。

クエリシンタックス *PSC?
ニーモニック *PSC?
省略形 *PSC?(省略形なし)

応答メッセージ <psc_flag>

・:SYSTem:HEADer設定ON/OFFに関わりなく、ヘッダ部はありません。
・<psc_flag> はパワーオンステータスクリアフラグの内容を示す数値で、＜NR1 数値応答データ＞フォーマットの 0 または 1 となります。

例 PRINT@DSO;"*PSC?"

3.1.12 *SRE コマンド /*SRE? クエリ(Service Request Enable)

***SRE コマンド**

サービスリクエストイネイブルレジスタの各ビットを設定します。

コマンドシンタックス　*SRE <mask_arg>
ニーモニック　*SRE
省略形　*SRE(省略形なし)

・<mask_arg> はサービスリクエストイネイブルレジスタ設定用のパラメータで＜数値プログラムデータ＞フォーマットの 0～255 の範囲の数値です。
・サービスリクエストイネイブルレジスタは各ビットに 1 つづつ意味が割り振られています。イネブルレジスタの対応するビットを 1 にしたときに対応する事象がイネブル(サービスリクエスト発生要因になる)、0 にしたときがディスエイブル(サービスリクエスト発生要因にならない)になります。

例　PRINT@DSO;"*SRE 255"

***SRE? クエリ**

サービスリクエストイネイブルレジスタの、現在の内容を読み出します。

クエリシンタックス　*SRE?
ニーモニック　*SRE?
省略形　*SRE?(省略形なし)

応答メッセージ　<mask>

・:SYSTem:HEADer 設定ON/OFF に関わりなく、ヘッダ部はありません。
・<mask> はサービスリクエストイネイブルレジスタの各ビットの設定を 2 のべき乗で重み付けした数値で＜NR1 数値応答データ＞フォーマットの 0 ～255 の範囲の数値です。

例　PRINT@DSO;"*SRE?"

備　考　サービスリクエストイネイブルレジスタ構成を表3.1.3 に示します。

表3.1.3 サービスリクエストイネイブルレジスタ構成

ビット	重み付け	意　味
7	128	- 本器では使用しないため、常に0
6	64	RQS- リクエストサービス
5	32	ESB- イベントステータス
4	16	MAV- メッセージアベイラブル
3	8	- 本器では使用しないため、常に0
2	4	GNG- GO/NOGOイベントステータス(ｵﾌﾟｼｮﾝ)
1	2	LCL- ローカルキューステータス
0	1	TRG- トリガイベントステータス

[注]ビット6 は *SRE コマンドのパラメータによる指定に関わらず、常に 0 とみなして応答データを作成します。但し、実際の動作としては、常にイネイブル状態とみなされることに注意して下さい。

3.1.13 *STB? クエリ(Status Byte)

本クエリでステータスバイトとマスタサマリステータスビット(MSSメッセージ)を読み出します。

クエリシンタックス　*STB?
　　ニーモニック　*STB?
　　省略形　*STB?(省略形なし)

応答メッセージ　<status>
・:SYSTem:HEADer 設定 ON/OFF に関わりなく、ヘッダ部はありません。
・<status> はステータスバイトレジスタの各ビットの値を 2 のべき乗で重み付けした数値で＜NR1 数値応答データ＞フォーマットの 0～255 の範囲の数値です。

例　PRINT@DSO;"*STB?"

備　考　ステータスバイトレジスタ構成を表3.1.4 に示します。

表3.1.4 ステータスバイトレジスタ構成

ビット	重み付け	意　味
7	128	- 本器では使用しないため、常に0
6	64	MSS- メッセージ サマリ ステータス
5	32	ESB- イベント ステータス
4	16	MAV- メッセージ アベイラブル
3	8	- 本器では使用しないため、常に0
2	4	GNG- GO/NOGOイベントステータス(ｵﾌﾟｼｮﾝ)
1	2	LCL- ローカルキュー ステータス
0	1	TRG- トリガイベントステータス

・*STB? クエリで、ステータスバイトレジスタの内容を読み取ると、IEEE 488.2 規格に従い、ビット 6 として RQS ではなく、MSS メッセージが使用されます。それ以外は、シリアルポールで読み出される値と同一となります。
・MSS メッセージは、ステータスバイトレジスタの内容をサービスリクエストイネイブルレジスタの内容でマスク(論理AND)して、ビット 6 以外の全ビットの論理 OR をとって求められます。
・RQS メッセージが基本的には、シリアルポールが実施されるまでステータスバイトレジスタに「こびりつく」のに対して、MSS メッセージはデバイスの内部ステータスをリアルタイムに反映します。

3.1.14 *TRG コマンド(Trigger)

本コマンドで本器をラン(RUN)状態にします。

コマンドシンタックス *TRG
ニーモニック *TRG
省略形 *TRG(省略形なし)

例 PRINT@DSO;"*TRG"

3.1.15 *RCL コマンド(Recall)

* RCL <register_number_arg> コマンドで本器の内部メモリにセーブしているセットアップファイルをリコールします。メモリカードにセーブしているファイルをリコールすることはできません。

コマンドシンタックス *RCL <register_number_arg>
ニーモニック *RCL
省略形 *RCL(省略形なし)
・<register_number_arg>はリコールすべきファイル番号指定用パラメータで<数値プログラムデータ>フォーマットの 0～98 の範囲の数値です。
・ローカル操作のセーブ/リコール動作で選択する SET000～SET098 の範囲のファイル名の下 3 桁の数字を指定します。

エラー処理
・<register_number_arg>が省略された場合には、コマンドエラーとし、エラーキューに以下のメッセージを格納します。
-109,"Missing parameter"
・<register_number_arg> を丸めて整数化した値が 99 以上の場合には実行エラーとし、エラーキューに以下のメッセージを格納します。
-222,"Data out of range"
・<register_number_arg> を丸めて整数化した値によって指定される番号のセットアップ ファイルが存在しない場合、実行エラーとし、エラーキューに以下のメッセージを格納します。
-256,"File name not found"

例 PRINT@DSO;"*RCL 15"

3.1.16 *SAV コマンド(Save)

*SAV <register_number_arg> コマンドで本器の内部メモリへにセットアップ情報をセーブします。

コマンドシンタックス　:SETup:SAVE <register_number>

ニーモニック　*SAV

省略形　*SAV(省略形なし)

・<register_number_arg> はセーブするファイル番号指定用パラメータで、＜数値プログラムデータ＞フォーマットの 10～98 の範囲の数値です。

・ローカル操作のセーブ/リコール動作で選択する SET000 ～ SET098 の範囲のファイル名の下 3 桁の数字を指定します。

エラー処理

・<register_number_arg> が省略された場合には、コマンドエラーとし、エラーキューに以下のメッセージを格納します。
-109,"Missing parameter"

・<register_number_arg> を丸めて整数化した値が 99 以上の場合には実行エラーとし、エラーキューに以下のメッセージを格納します。
-222,"Data out of range"

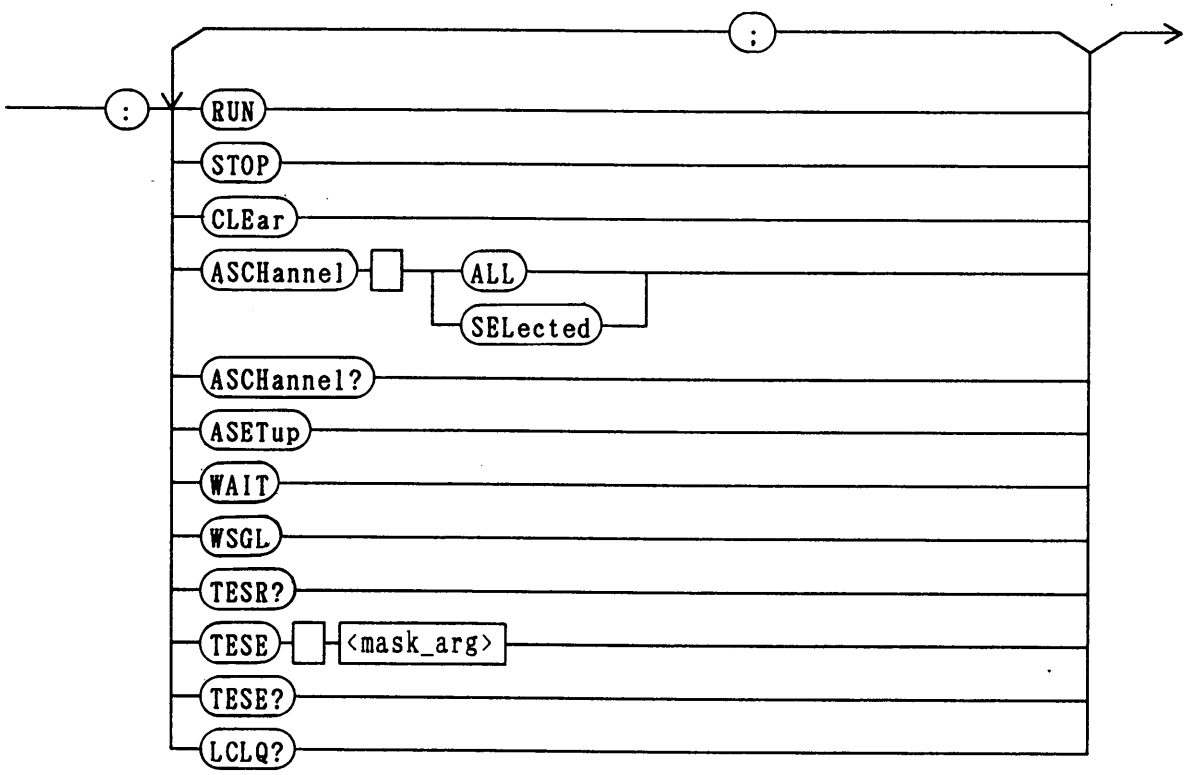
・:SETup:SECurity 設定=ON の状態で、かつ <register_number_arg> を丸めて整数化した値によって指定される番号のセットアップファイルが既にメモリ上に存在した場合、実行エラーとし、エラーキューに以下のメッセージを格納します。
-257,"File name error"

例　PRINT@DSO;"*SAV 15"

3.2 ルート(ROOT)レベルコマンド

ルートレベルコマンドは本器のラン/ストップの制御やオートセットアップの実行などの基本的な機能の設定、確認、および動作の実行を行うためのコマンドです。
ルートレベルコマンドは、他の親コマンド群とは異なり、サブコマンドはありません。

図3.2.1 ルートレベルコマンド全体像



3.2.1 RUN コマンド

:RUN コマンドで本器をラン状態にします。但し、掃引方式(:TRIGger:SWEep)設定=SINGle の場合は、波形測定リセット(単掃引起動)コマンドとして動作します。

コマンドシンタックス :RUN
ニーモニック RUN
省略形 RUN(省略形なし)

例 PRINT@DSO;":RUN"

備　考 ・ローカル動作の場合、[RUN/STOP]はラン/ストップのトグル切り換えキーとして動作します。
・リモートの RUN コマンドはラン状態指定コマンドとして動作します。

注　意 既にラン状態になっている場合、RUN コマンドを受信しても、ラン状態のままで、ストップしません。

3.2.2 STOP コマンド

:STOP コマンドで本器をストップ状態にします。

コマンドシンタックス :STOP
ニーモニック STOP
省略形 STOP(省略形なし)

例 PRINT@DSO;":STOP"

備　考 ・ローカル動作の場合、[RUN/STOP]はラン/ストップのトグル切り換えキーとして動作します。
・リモートの STOP コマンドはストップ状態指定コマンドとして動作します。

注　意 既にストップ状態になっている場合、STOP コマンドを受信しても、ストップ状態のままで、ラン状態にはなりません。

3.2.3 CLEar コマンド

:CLEar コマンドでパーシスタンス表示をクリアします。

コマンドシンタックス :CLEar
ニーモニック CLEAR
省略形 CLE

例 PRINT@DSO;":CLEAR"
PRINT@DSO;":CLE"

3.2.4 ASCHannel コマンド/ ASCHannel? クエリ

:ASCHannel <auto_setup_ch_arg>

本コマンドでオートセットアップモードを設定します。

コマンドシンタックス :ASCHannel <auto_setup_ch_arg>
ニーモニック ASCHANNEL
省略形 ASCH
・<auto_setup_ch_arg> はオートセットアップモード設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・ALL、SELected のいずれかとします。

例
```
PRINT@DSO;":ASCHANNEL SELECTED"
PRINT@DSO;":ASCH SEL"
```

:ASCHannel?

本クエリで現在のオートセットアップモードの設定を読み出します。

クエリシンタックス :ASCHannel?
ニーモニック ASCHANNEL?
省略形 ASCH?

応答メッセージ :ASCHannel <auto_setup_ch>
・「:ASCHannel 」部分は :SYSTem:HEADer設定＝OFF の場合には省略します。
・<auto_setup_ch> は＜文字応答データ＞フォーマットのニーモニックです。
・ALL、SELected のいずれかとします。

3.2.5 ASETup コマンド

:ASETup コマンドでオートセットアップを行ないます。

コマンドシンタックス :ASETup
ニーモニック ASETUP
省略形 ASET

例
```
PRINT@DSO;":ASETUP"
PRINT@DSO;":ASET"
```

3.2.6 WAIT コマンド

:WAIT コマンドで掃引方式(:TRIGger:SWEep)設定＝SINGle の場合の測定完了を待ちます。

コマンドシンタックス　:WAIT
　　ニーモニック　WAIT
　　省略形　　WAIT(省略形なし)

例　　PRINT@DSO;":WAIT"

3.2.7 WSGL コマンド

:WSGL コマンドで掃引方式(:TRIGger:SWEep)設定を SINGle とした後、ラン状態とします(単掃引での測定を 1 回行う)。

コマンドシンタックス　:WSGL
　　ニーモニック　WSGL
　　省略形　　WSGL(省略形なし)

例　　PRINT@DSO;":WSGL"

3.2.8 TESR? クエリ

:TESR? クエリでトリガイベントステータスレジスタ(単掃引での測定が終了したか否かのステータス)を読み出します(読み出した後、クリア)。

クエリシンタックス　:TESR?
　　ニーモニック　TESR?
　　省略形　　TESR?(省略形なし)

例　　PRINT@DSO;":TESR?"

応答メッセージ　　<status>
・:TESR? クエリに対する応答データは、:SYSTem:HEADer 設定 ON/OFF 設定に関わらず、ヘッダ部を含みません。
・<status> はステータスを表す数値で ＜NR1数値応答データ＞フォーマットとします。
・シングル測定が終了している場合は 1、それ以外では 0 とします。

注　意　　読み出した後、クリアします。

3.2.9 TESE コマンド/ TESE? クエリ

:TESE <mask_arg>

本コマンドでトリガイベントステータスイネーブルレジスタ(トリガイベントステータスレジスタのサマリメッセージをステータスバイトに反映させるか否か)を設定します。

コマンドシンタックス　:TESE <mask_arg>
　　ニーモニック　TESE
　　省略形　TESE(省略形なし)
　　<mask_arg> はトリガイベントステータスイネーブルレジスタ設定用のパラメータで、＜数値プログラムデータ＞フォーマットとし、0～255 の範囲の数値です。 但し、内部的には <mask_arg>&1 の値を登録します(ビット 0 のみ使用)。

例　PRINT@DSO;":TESE 1"

:TESE?

本クエリで現在のトリガイベントステータスイネーブルレジスタの設定を読み出します。

クエリシンタックス　:TESE?
　　ニーモニック　TESE?
　　省略形　TESE?(省略形なし)

応答メッセージ　:TESE <mask>
- :TESE? クエリに対する応答は :SYSTem:HEADer設定 ON/OFF に関わらず、ヘッダ部を含みません。
- <mask> は設定値を表す数値で＜NR1 数値応答データ＞フォーマットとします。
- イネイブルの場合は 1、それ以外では 0 とします。

3.2.10 LCLQ? クエリ

:LCLQ? クエリにより、ローカルキュー(リモート状態からローカル状態への移行が発生したか否か)を読み出すことが可能です(読み出して後クリア)。

クエリシンタックス　:LCLQ?
　　ニーモニック　LCLQ?
　　省略形　LCLQ?(省略形なし)

例　PRINT@DSO;":LCLQ?"

応答メッセージ　<status>
- :LCLQ? クエリに対する応答は :SYSTem:HEADer 設定 ON/OFF に関わらず、ヘッダ部を含みません。
- <status> はステータスを表す数値で＜NR1 数値応答データ＞フォーマットとします。
- 電源投入あるいは前回の :LCLQ? クエリによる読み出しや、*CLS コマンドによるクリアの後に、リモート状態からローカル状態への移行が発生している場合は 1、それ以外では 0 とします。

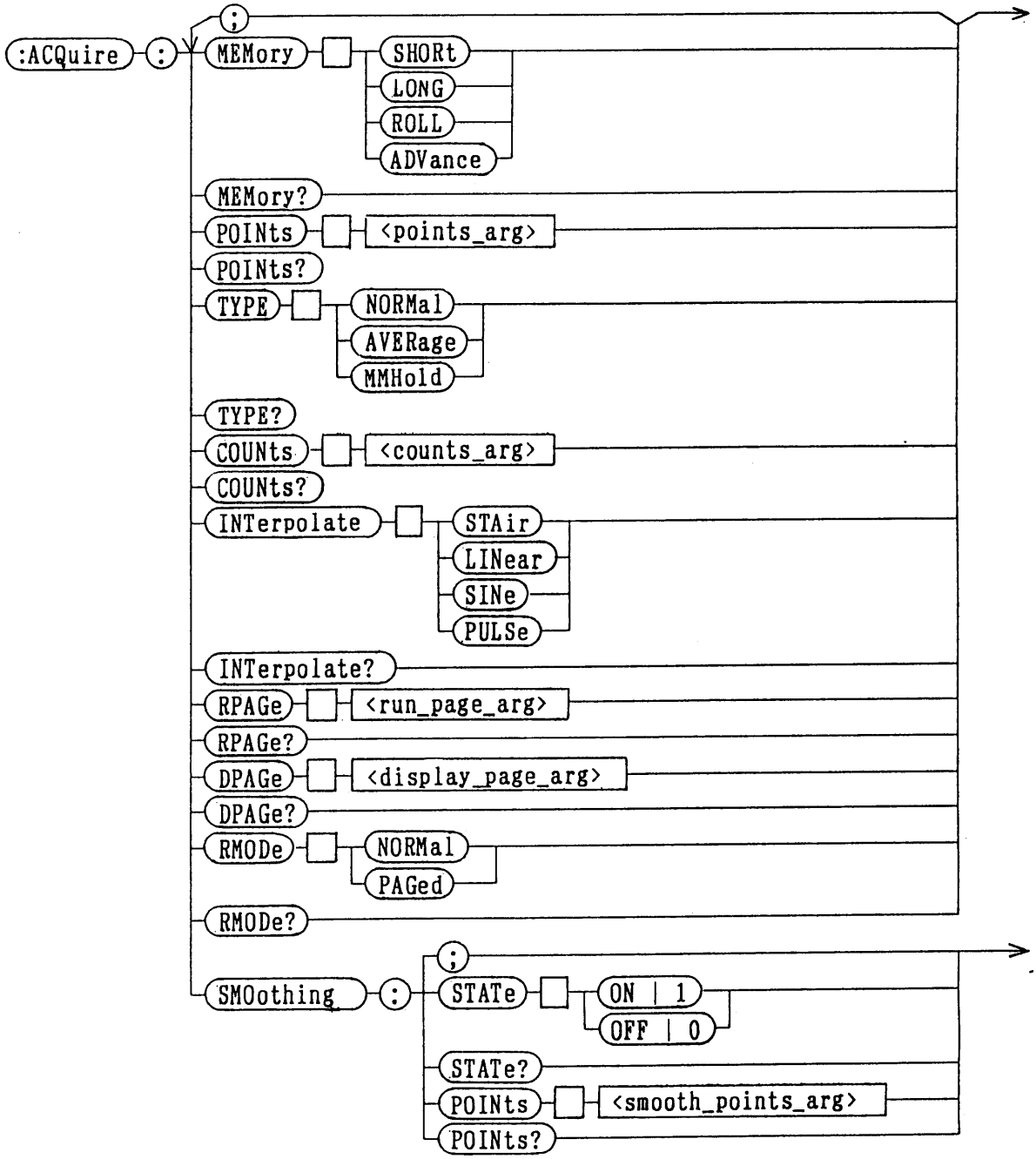
注　意　読み出した後、クリアします。

3.3 ACQuire 関連コマンド

:ACQuire:<item>……　と指定すると、ストレージメニューを対象とする各種設定と確認が可能です。
実際の設定項目は <item> 部のヘッダにより指定されます。

コマンド シンタックス　:ACQuire:<item>……
　　ニーモニック　　ACQUIRE
　　省略形　　ACQ

図3.3.1　ACQuire 関連コマンド全体像



3.3.1 MEMory コマンド/ MEMory? クエリ

:ACQuire:MEMory <memory_mode_arg>

本コマンドでメモリ使用種別を設定します。

コマンドシンタックス　:ACQuire:MEMory <memory_mode_arg>
　　ニーモニック　MEMORY
　　省略形　MEM

・<memory_mode_arg> はメモリ使用種別設定用パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・SHORt、LONG、ROLL、ADVance のいずれかとします。

エラー処理　メモリアドバンス(オプション)が実装されていない場合、ADVance 指定はコマンドエラーとなり、エラーキューに以下のメッセージを格納します。
-141,"Invalid character data"

例
```
PRINT@DSO;":ACQUIRE:MEMORY SHORT"
PRINT@DSO;":ACQ:MEM shor"
```

:ACQuire:MEMory?

本クエリで、現在のメモリ使用種別の設定値を読み出します。

クエリシンタックス　:ACQuire:MEMory?
　　ニーモニック　MEMORY?
　　省略形　MEM?

応答メッセージ　:ACQuire:MEMory <memory_mode>

・「:ACQuire:MEMory」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<memory_mode> は現在のメモリ使用種別を示す、＜文字応答データ＞フォーマットのニーモニックです。
・SHORt、LONG、ROLL、ADVance のいずれかとなります。

3.3.2 POINts コマンド/ POINts? クエリ

:ACQuire:POINts <memory_length_arg>

本コマンドでメモリ長を設定します。

コマンドシンタックス :ACQuire:POINts <memory_length_arg>
ニーモニック POINTS
省略形 POIN

・<memory_length_arg> はメモリ長設定用のパラメータで、<数値プログラムデータ>フォーマットの数値です。
・設定値は、丸めにより、設定可能なメモリ長設定の値に飛びつきます。
・サフィックス単位は許しません。
・メモリ長として指定可能な設定値は、メモリ使用種別により異なります。
・メモリ長はバイト単位で設定します。
・エンベロープモードのとき、メモリ長が 64k 以上の値に設定されると、エンベロープモードは自動的に解除されます。

例
```
PRINT@DSO;":ACQuire:POINTS 512"
PRINT@DSO;":ACQ:POIN 32768"
```

:ACQuire:POINts?

本クエリで現在のメモリ長の設定値を読み出します。

クエリシンタックス :ACQuire:POINts?
ニーモニック POINTS?
省略形 POIN?

応答メッセージ :ACQuire:POINts <memory_length>

・「:ACQuire:POINts」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<memory_length> は<NR1 数値応答データ>フォーマットの数値で、現在のメモリ長設定を表します。

3.3.3 TYPE コマンド/ TYPE? クエリ

:ACQuire:TYPE <acq_type_arg>

本コマンドで波形処理モードを設定します。

エラー処理
- TYPE コマンド/クエリともにメモリ使用種別(:ACQuire:MEMory設定)が SHORT 以外の場合には実行エラーとなり、エラーキューに以下のメッセージを格納します。
  -221,"Setting conflict"
- メモリ使用種別 (:ACQuire:MEMory 設定) が SHORT でかつ波形演算の演算子 (:DISPlay:CALCulate:FUNCtion 設定) がスペクトル (SPECtrum) の場合にマックスホールド (MMHold) を指定すると実行エラーとなり、エラーキーに以下のメッセージを格納します。
  -221,"Setting conflict"

コマンドシンタックス　:ACQuire:TYPE <acq_type_arg>
ニーモニック　TYPE
省略形　TYPE (省略形なし)

- <acq_type_arg> は波形処理モード設定用のパラメータで、<文字プログラムデータ>フォーマットのニーモニックです。
- NORMal、AVERage、MMHold のいずれかとします。
- MMHold はマックスホールド指定を意味します。

例
```
PRINT@DSO;":ACQUIRE:TYPE NORMAL"
PRINT@DSO;":ACQ:TYPE AVER"
```

:ACQuire:TYPE?

本クエリで現在の波形処理モードの設定値を読み出します。

クエリシンタックス　:ACQuire:TYPE?
ニーモニック　TYPE?
省略形　TYPE? (省略形なし)

応答メッセージ　:ACQuire:TYPE <acq_type>

- 「:ACQuire:TYPE」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
- <acq_type> は<文字応答データ>フォーマットのニーモニックです。
- NORMal、AVERage、MMHold のいずれかです。

3.3.4 COUNts コマンド/ COUNts? クエリ

:ACQuire:COUNts <avg/max_counts_arg>

本コマンドでアベレージ または マックスホールド回数を設定します。

エラー処理　COUNts コマンド/クエリともにメモリ使用種別(:ACQuire:MEMory設定)が SHORT 以外の場合には実行エラーとなり、エラーキューに以下のメッセージを格納します。
-221,"Setting conflict"

コマンドシンタックス　:ACQuire:COUNts <avg/max_counts_arg>
ニーモニック　COUNTS
省略形　COUN

・<avg/max_counts_arg> はアベレージあるいはマックスホールド回数設定用パラメータで、＜数値プログラムデータ＞フォーマットの数値です。
・設定値は、丸めにより整数値に変換し、設定可能範囲外の場合には、設定可能な値に飛びつきます。

例
```
PRINT@DSO;":ACQUIRE:COUNTS 9999"
PRINT@DSO;":ACQ:COUN 1"
```

:ACQuire:COUNts?

本クエリで、現在のアベレージ または マックスホールド回数の設定値を読み出します。

クエリシンタックス　:ACQuire:COUNts?
ニーモニック　COUNTS?
省略形　COUN?

応答メッセージ　:ACQuire:COUNts <avg/max_counts>

・「:ACQuire:COUNts」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<avg/max_counts> は、現在のアベレージあるいはマックスホールド回数を表す＜NR1 数値応答データ＞フォーマットの数値です。

3.3.5 INTerpolate コマンド/ INTerpolate? クエリ

:ACQuire:INTerpolate <interpolate_arg>

本コマンドで補間方式を設定します。

コマンドシンタックス　:ACQuire:INTerpolate <interpolate_arg>
　　ニーモニック　INTERPOLATE
　　省略形　INT
　　・<interpolate_arg> は補間方式設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
　　・STAir、LINear、SINE、PULSe のいずれかとします。
例　PRINT@DSO;":ACQUIRE:INTERPOLATE STAIR"
　　PRINT@DSO;":ACQ:INT LIN"

:ACQuire:INTerpolate?

本クエリで、現在の補間方式の設定を読み出します。

クエリシンタックス　:ACQuire:INTerpolate?
　　ニーモニック　INTERPOLATE?
　　省略形　INT?

応答メッセージ　:ACQuire:INTerpolate <interpolate>
　　・「:ACQuire:INTerpolate」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<interpolate> は＜文字応答データ＞フォーマットのニーモニックで、現在の補間方式を示します。
　　・STAir、LINear、SINE、PULSe のいずれかとなります。

3.3.6 RPAGe コマンド/ RPAGe? クエリ

:ACQuire:RPAGe <run_page_arg>

本コマンドでランページ数を設定します。

エラー処理
- RPAGe コマンド/クエリともに メモリアドバンス(ｵﾌﾟｼｮﾝ)を実装していない場合にはコマンドエラーとなり、エラーキューに以下のメッセージを格納します。
  -113,"Undefined header"
- RPAGe コマンド/クエリともにメモリ使用種別(:ACQuire:MEMory設定)がメモリアドバンス(ADVance)以外の場合には実行エラーとなり、エラーキューに以下のメッセージを格納します。
  -221,"Setting conflict"

コマンドシンタックス :ACQuire:RPAGe <run_page_arg>
ニーモニック RPAGE
省略形 RPAG
- <run_page_arg> はランページ数設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値です。
- 設定値は、丸めにより整数値に変換し、設定可能範囲外の場合には、設定可能な値に飛びつきます。
- DPAGe設定値＞RPAGe設定値となる場合には DPAGe設定値=RPAGe設定値となるように、DPAGe設定値を変更します。

例
```
PRINT@DSO;":ACQUIRE:RPAGE 16"
PRINT@DSO;":ACQ:RPAG 1"
```

:ACQuire:RPAGe?

本クエリで、現在のランページ数の設定値を読み出します。

クエリシンタックス :ACQuire:RPAGe?
ニーモニック RPAGE?
省略形 RPAG?

応答メッセージ :ACQuire:RPAGe <run_page>
- 「:ACQuire:RPAGe」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
- <run_page> は、＜NR1 数値応答データ＞フォーマットの数値で、現在のランページ設定値を示します。

3.3.7 DPAGe コマンド/ DPAGe? クエリ

:ACQuire:DPAGe <display_page_arg>

本コマンドで表示ページ数を設定します。

エラー処理
- DPAGe コマンド/クエリともに、メモリアドバンス(ｵﾌﾟｼｮﾝ)を実装していない場合にはコマンドエラーとなり、エラーキューに以下のメッセージを格納します。
  -113,"Undefined header"
- DPAGe コマンド/クエリともにメモリ使用種別(:ACQuire:MEMory設定)がメモリアドバンス(ADVance)以外の場合には実行エラーとなり、エラーキューに以下のメッセージを格納します。
  -221,"Setting conflict"
- DPAGe設定値＞RPAGe設定値となる場合には実行エラーとなり、エラーキューに以下のメッセージを格納します。
  -221,"Setting conflict"

コマンドシンタックス　:ACQuire:DPAGe <display_page_arg>
ニーモニック　DPAGE
省略形　DPAG

- <display_page_arg> は表示ページ数設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値です。
- 設定値は、丸めにより整数値に変換し、設定可能範囲外の場合には、設定可能な値に飛びつきます。

例
```
PRINT@DSO;":ACQUIRE:DPAGE 1"
PRINT@DSO;":ACQ:DPAG 1"
```

:ACQuire:DPAGe?

本クエリで、現在の表示ページ数の設定値を読み出します。

クエリシンタックス　:ACQuire:DPAGe?
ニーモニック　DPAGE?
省略形　DPAG?

応答メッセージ　:ACQuire:DPAGe <display_page>

- 「:ACQuire:DPAGe」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
- <display_page> は ＜NR1 数値応答データ＞フォーマットの数値で、現在の表示ページ数を示します。

3.3.8 RMODe コマンド/ RMODe? クエリ

:ACQuire:RMODe <roll_mode_arg>

本コマンドで ロールモードを設定します。

コマンドシンタックス　:ACQuire:RMODe <roll_mode_arg>
ニーモニック　RMODE
省略形　RMOD

・<roll_mode_arg>はロールモード設定用のパラメータで、<文字プログラムデータ>フォーマットのニーモニックです。
・NORMal、PAGed のいずれかとします。

エラー処理

・RMODe コマンド/クエリともに、メモリ使用種別(:ACQuire:MEMory 設定)がロール(ROLL)以外の場合は実行エラーとなり、エラーキューに以下のメッセージを格納します。
-221,"Setting conflict"
・DS-9122 の場合、増設メモリ1M(ｵﾌﾟｼｮﾝ)が実装されていない場合はコマンドエラーとなり、エラーキューに以下のメッセージを格納します。
-141,"Invalid charactor data"

:ACQuire:RMODe?

本クエリで 現在のロールモードの設定状態を読み出します。

クエリシンタックス　:ACQuire:RMODe?
ニーモニック　RMODE?
省略形　RMOD?

応答メッセージ　:ACQuire:RMODe <roll_mode>

・「:ACQuire:RMODe」部分は:SYSTem:HEADer 設定=OFF の場合には省略します。
・<roll_mode>は<文字応答データ>フォーマットのニーモニックです。
・NORMal、PAGed のいずれかになります。

注　意　ロールモードを PAGED に設定後 WSGL コマンドを実行するとコマンドエラーとなり、エラーキューに以下のメッセージを格納します。
-221,"Setting conflict"

3.3.9 SMOothing/ SMOothing?

:ACQuire:SMOothing:<item>･･･と指定すると、スムージング機能を対象とする各種設定が可能です。
実際の設定項目は SMOothing サブコマンドの後ろに続く<item>部のヘッダにより指定されます。

コマンドシンタックス :ACQuire:SMOothing:<item>･･･
ニーモニック SMOOTHING
省略形 SMO

3.3.9.1 STATe コマンド/ STATe? クエリ

:ACQuire:SMOothing:STATe <mode_arg>

本コマンドで スムージング機能の ON/OFF を設定します。

コマンドシンタックス :ACQuire:SMOothing:STATe <mode_arg>
ニーモニック STATE
省略形 STAT

・<mode_arg>はスムージングの ON/OFF 設定用のパラメータで、<文字プログラムデータ>フォーマットの ON または OFF、<数値プログラムデータ>フォーマットの数値の何れかとします。
・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

エラー処理

・STATeコマンド/クエリともに、メモリ使用種別(:ACQuire:MEMory 設定)がロング(LONG) または ロール(ROLL)で かつロールモード(:ACQuire:RMODe 設定)がページドロール(PAGed)の場合は実行エラーとなり、エラーキューに以下のメッセージを格納します。
-221,"Setting conflict"

:ACQuire:SMOothing:STATe?

本クエリで 現在のスムージング機能の設定状態を読み出します。

クエリシンタックス :ACQuire:SMOothing:STATe?
ニーモニック STATE?
省略形 STAT?

応答メッセージ :ACQuire:SMOothing:STATe <mode>

・「:ACQuire:SMOothing:STATe」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode>は<NR1 数値応答データ>フォーマットとし 0 または 1 の 1 桁の数字とします。
・0 が OFF、1 が ON を意味します。

3.3.9.2 POINts コマンド/ POINts? クエリ

:ACQuire:SMOothing:POINts <smooth_points_arg>

本コマンドで スムージング回数を設定します。

コマンドシンタックス :ACQuire:SMOothing:POINts <smooth_points_arg>
ニーモニック POINTS
省略形 POIN

・<smooth_points_arg>はスムージング回数設定用のパラメータで、<数値プログラムデータ>フォーマットの数値です。
・設定値は、丸めにより整数値に変換し、設定範囲外の場合には設定可能な値に飛びつきます。

エラー処理

・POINts コマンド/クエリともに、メモリ使用種別(:ACQuire:MEMory 設定)がロング(LONG) または ロール(ROLL)で かつロールモード(:ACQuire:RMODe 設定)がページドロール(PAGed)の場合は実行エラーとなり、エラーキューに以下のメッセージを格納します。
-221,"Setting conflict"

:ACQuire:SMOothing:POINts?

本クエリで 現在のスムージング回数の設定値を読み出します。

クエリシンタックス :ACQuire:SMOothing:POINts?
ニーモニック POINts?
省略形 POIN?

応答メッセージ :ACQuire:SMOothing:POINts <smooth_points>

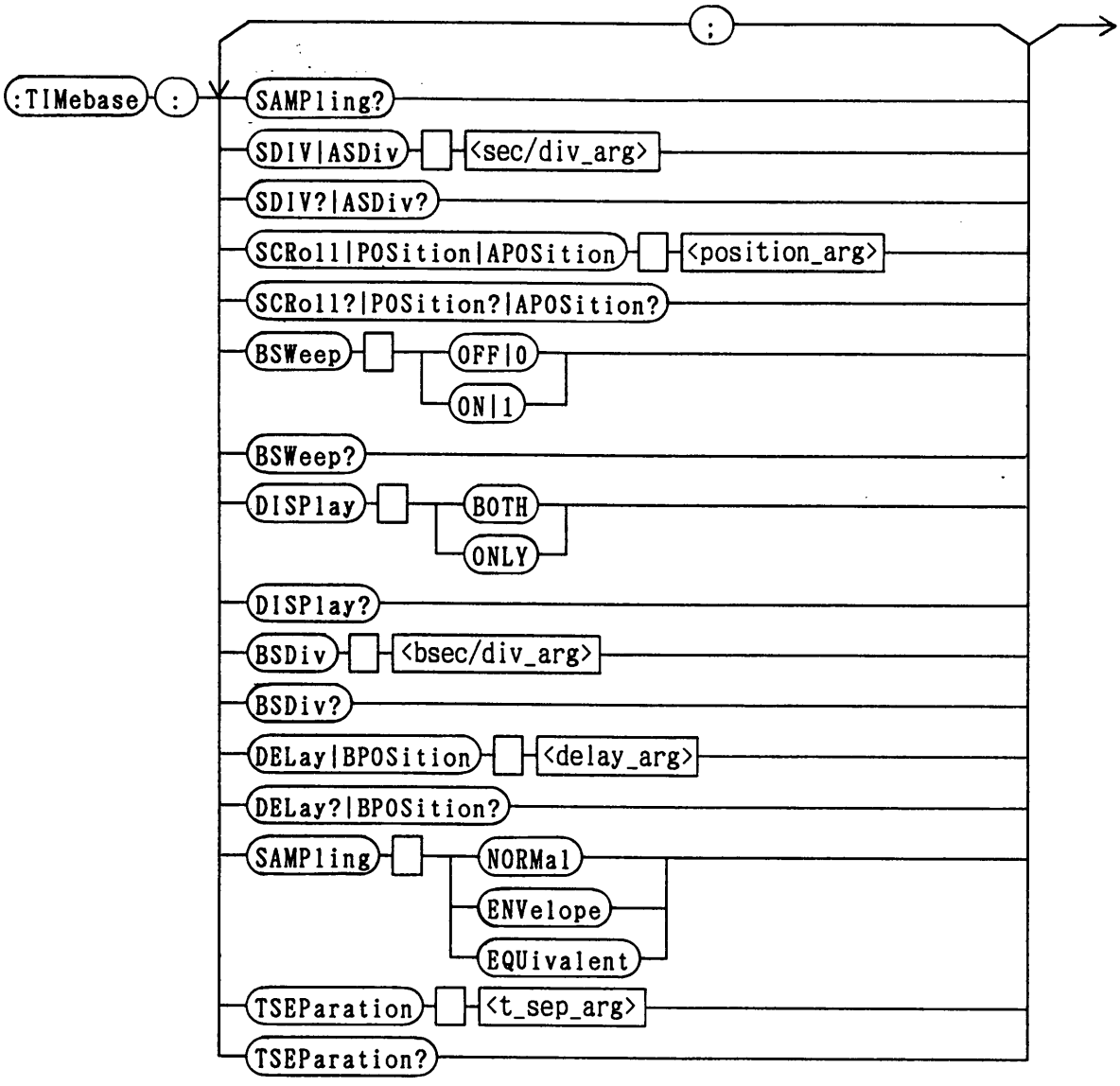
・「:ACQuire:SMOothing:POINts」部分は :SYSTem:HEADer 設定=OFF の場合には省略します。
・<smooth_points>は、現在のスムージング回数を表す<NR1 数値応答データ>フォーマットの数値です。

3.4 TIMebase 関連コマンド

:TIMebase:<item>……　と指定すると、時間軸を対象とする各種設定が可能です。
実際の設定項目は <item> 部のヘッダにより指定されます。

コマンドシンタックス　:TIMebase:<item>……
　　ニーモニック　TIMEBASE
　　省略形　　TIM

図3.4.1 TIMebase 関連コマンド全体像



3.4.1 SAMPling コマンド/ SAMPling? クエリ

:TIMebase:SAMPling <sampling_arg>

本コマンドでサンプリングモードを設定します。

コマンドシンタックス　:TIMebase:SAMPling <sampling_arg>
　　ニーモニック　SAMPLING
　　省略形　SAMP
・<sampling_arg> はサンプリングモード設定用のパラメータで ＜文字プログラムデータ＞フォーマットのニーモニックです。
・NORMal、ENVelope、EQUivalent のいずれかとします。

エラー処理　**DS-9121/DS-9122**
・EQUivalent 設定はメモリ使用種別(:ACQuire:MEMory)設定=SHORt モードでのみ有効で、それ以外の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
　-221,"Setting conflict"
・メモリ使用種別 (:ACQuire:MEMory 設定) が SHORt でかつ波形演算の演算子 (:DISPlay:CALCulate:FUNCtion 設定) がスペクトル (SPECtrum) の場合に ENVelope を指定すると実行エラーとなり、エラーキューに以下のメッセージを格納します。
　-221,"Setting conflict"

DS-9122 のみ
次のように設定されているとき、エンベロープが実行されると実行エラーとなり、以下のメッセージをエラーキューに格納します。
メモリ使用種別:ロング
アクイジションチャネル: CH1&2 または CH1_only
POINTs コマンドによるメモリ長の設定:64k 以上
-221,"Setting conflict"

例
```
PRINT@DSO;":TIMEBASE:SAMPLING NORMAL"
PRINT@DSO;":TIM:SAMP EQU"
```

:TIMebase:SAMPling?

本クエリで現在のサンプリングモード設定を読み出します。

クエリシンタックス　:TIMebase:SAMPling?
　　ニーモニック　SAMPLING?
　　省略形　SAMP?

応答メッセージ　:TIMebase:SAMPling <sampling>
・「:TIMebase:SAMPling」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<sampling> は＜文字プログラムデータ＞フォーマットのニーモニックです。
・NORMal、ENVelope、EQUivalent のいずれかとします。

3.4.2 SDIV コマンド/ SDIV? クエリ

:TIMebase:SDIV <sec/div_arg>

本コマンドで時間軸(SEC/DIV)を設定します。設定範囲は次の通りです。
DS-9121 : 5ns～50s(1-2-5 ｼｰｹﾝｽ)
DS-9122 : 2ns～20s(1-2-5 ｼｰｹﾝｽ)
但し、メモリ使用種別がロールか否かにより設定可能範囲が異なります。
ASDiv 表記も同一動作とします。

コマンドシンタックス :TIMebase:SDIV <sec/div_arg>
:TIMebase:ASDiv <sec/div_arg>
ニーモニック SDIV または ASDiv
省略形 SDIV または ASD

・<sec/div_arg> は時間軸設定用のパラメータで＜数値プログラムデータ＞フォーマットを基本とし、サフィックス部も有効とします。
・デフォルトのサフィックス乗数は×10^0、デフォルトのサフィックス単位は s とします。
・設定値は、丸めを行ってから評価します。1-2-5 ｼｰｹﾝｽでない設定値も、それだけではエラーにはしません。
但し、＜数値プログラムデータ＞、サフィックス部としてのフォーマットチェックの段階でコマンドエラーとなった場合を除きます。
・<sec/div_arg> の丸めの方法は以下の通りです。

DS-9121 の場合

<sec/div_arg>	≦5E-9(5ns)	→ 5ns/div
5E-9<	≦10E-9(10ns)	→ 5ns/div
｜	｜	
50E+0<		→ 50s/div

DS-9122 の場合

<sec/div_arg>	≦2E-9(2ns)	→ 2ns/div
2E-9<	≦5E-9(5ns)	→ 2ns/div
｜	｜	
20E+0<		→ 20s/div

例
```
PRINT@DSO;":TIMEBASE:SDIV 100US"
PRINT@DSO;":TIM:ASD 1.00E-4"
```

:TIMebase:SDIV?

本クエリで現在の時間軸設定値を読み出します。

クエリシンタックス :TIMebase:SDIV?
:TIMebase:ASDiv?
ニーモニック SDIV? または ASDiv?
省略形 SDIV? または ASD?

応答メッセージ :TIMebase:SDIV <sec/div>
・「:TIMebase:SDIV」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・ASDiv? クエリに対する応答の場合でも、応答メッセージのヘッダ部は SDIV とします。
・<sec/div> は＜NR3 数値応答データ＞フォーマットとします。

注 意 応答データは、単位を s とみなして計算、作成しますが、サフィックス部分は応答メッセージとしては送信しません。

3.4.3 SCRoll コマンド/ SCRoll? クエリ

:TIMebase:SCRoll <position_arg>

本コマンドでスクロール時間を設定します。設定範囲や分解能は、その時点の時間軸設定により異なります。POSition と APOSition 表記も同一動作とします。

コマンドシンタックス :TIMebase:SCRoll <position_arg>
:TIMebase:POSition <position_arg>
:TIMebase:APOSition <position_arg>

ニーモニック SCROLL, POSITION または APOSITION

省略形 SCR, POS または APOS

・<position_arg> はスクロール時間設定用のパラメータで ＜数値プログラムデータ＞フォーマットを基本とし、サフィックス部も有効とします。
・デフォルトのサフィックス乗数は×10^0、デフォルトのサフィックス単位は s とします。

例

```
PRINT@DSO;":TIMEBASE:SCROLL 100US"
PRINT@DSO;":TIM:APOS 1.00E-4"
```

:TIMebase:SCRoll?

本クエリで現在のスクロール時間設定値を読み出します。POSition? と APOSition? 表記も同一動作とします。

クエリシンタックス :TIMebase:SCRoll?
:TIMebase:POSition?
:TIMebase:APOSition?

ニーモニック SCROLL?, POSITION? または APOSITION?

省略形 SCR?, POS? または APOS?

応答メッセージ :TIMebase:SCRoll <position>

・「:TIMebase:SCRoll」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・POSition? または APOSition? クエリに対する応答の場合でも、応答メッセージのヘッダ部は SCRoll とします。
・<position> は＜NR3 数値応答データ＞フォーマットとします。

注 意 応答データは、単位を s とみなして計算、作成しますが、サフィックス部分は応答メッセージとしては送信しません。

3.4.4 BSWeep コマンド/ BSWeep? クエリ

:TIMebase:BSWeep <mode_arg>

本コマンドで B 掃引 ON/OFF を設定します。

エラー処理　　:TIMebase:BSWeep コマンド/クエリとも、メモリ使用種別(:ACQuire:MEMory)設定=SHORt モード でのみ有効で、それ以外の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TIMebase:BSWeep <mode_arg>
ニーモニック　BSWEEP
省略形　　BSW

・<mode_arg> は B 掃引 ON/OFF 設定用のパラメータで、<文字プログラムデータ>フォーマットの ON または OFF、<数値プログラムデータ>フォーマットの数値の何れかとします。
・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

例　　PRINT@DSO;":TIMEBASE:BSWEEP 1"
PRINT@DSO;":TIM:BSW ON"

:TIMebase:BSWeep?

本クエリで現在の B 掃引の設定を読み出します。

クエリシンタックス　:TIMebase:BSWeep?
ニーモニック　BSWEEP?
省略形　　BSW?

応答メッセージ　　:TIMebase:BSWeep <mode>

・「:TIMebase:BSWeep」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は<NR1 数値応答データ>フォーマットとし、0 または 1 の 1 桁の数字とします。
・0 が OFF、1 が ON を意味します。

3.4.5 DISPlay コマンド/ DISPlay? クエリ

:TIMebase:DISPlay <display_arg>

本コマンドで B 掃引 ON の場合のディスプレイ BOTH/ONLY を設定します。

エラー処理　　:TIMebase:DISPlay コマンド/クエリとも、メモリ使用種別(:ACQuire:MEMory)設定＝SHORt かつ B 掃引(:TIMebase:BSWeep)設定＝ON でのみ有効で、それ以外の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TIMebase:DISPlay <display_arg>
ニーモニック　DISPLAY
省略形　　DISP
・<display_arg> はディスプレイ設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・BOTH または ONLY のいずれかとします。

例　　PRINT@DSO;":TIMEBASE:DISPLAY ONLY"
PRINT@DSO;":TIM:DISP BOTH"

:TIMebase:DISPlay?

本クエリで現在のディスプレイ設定を読み出します。

クエリシンタックス　:TIMebase:DISPlay?
ニーモニック　DISPLAY?
省略形　　DISP?

応答メッセージ　　:TIMebase:DISPlay <display>
・「:TIMebase:DISPlay」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<display> は＜文字プログラムデータ＞フォーマットのニーモニックです。
・BOTH または ONLY のいずれかとします。

3.4.6 BSDiv コマンド/ BSDiv? クエリ

:TIMebase:BSDiv <sec/div_arg>

本コマンドで B 掃引時間軸(B SEC/DIV)を設定します。設定範囲は次の通りです。
DS-9121 : 5ns～50s(1-2-5 ｼｰｹﾝｽ)
DS-9122 : 2ns～20s(1-2-5 ｼｰｹﾝｽ)

注　　意

・A 掃引時間軸より遅い時間軸に設定すると、A 掃引時間軸が連動します。
・メモリ使用種別がロールか否かにより設定可能範囲が異なります。

エラー処理　　:TIMebase:BSDiv コマンド/クエリとも、メモリ使用種別(:ACQuire:MEMory)設定=SHORt かつ、B 掃引(:TIMebase:BSWeep)設定＝ON でのみ有効で、それ以外の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TIMebase:BSDiv <sec/div_arg>
ニーモニック　BSDIV
省略形　BSD

・<sec/div_arg> は B 掃引時間軸設定用のパラメータで、＜数値プログラムデータ＞フォーマットを基本とし、サフィックス部も有効とします。
・デフォルトのサフィックス乗数は×10⁰、デフォルトのサフィックス単位は s とします。
・設定値は、丸めを行ってから評価します。1-2-5 シーケンスでない設定値も、それだけではエラーにはしません。
但し、＜数値プログラムデータ＞、サフィックス部としてのフォーマットチェックの段階でコマンドエラーとなった場合を除きます。
<sec/div_arg> の丸めの方法は以下の通りです(ASDiv と同じ)。

DS-9121 の場合

<sec/div_arg>		
	≦ 5E-9(2ns)	→ 5ns/div
5E-9<	≦10E-9(10ns)	→ 10ns/div
|	|	
50E+0<		→ 50s/div

DS-9122 の場合

<sec/div_arg>		
	≦ 2E-9(2ns)	→ 2ns/div
2E-9<	≦ 5E-9(5ns)	→ 5ns/div
|	|	
20E+0<		→ 20s/div

例　　PRINT@DSO;":TIMEBASE:BSDIV 100US"
PRINT@DSO;":TIM:BSD 1.00E-4"

:TIMebase:BSDiv?

本クエリで、現在の B 掃引時間軸設定値を読み出します。

クエリシンタックス　:TIMebase:BSDiv?
ニーモニック　BSDIV?
省略形　BSD?

応答メッセージ　:TIMebase:BSDiv <sec/div>
・「:TIMebase:BSDiv」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<sec/div> は ＜NR3 数値応答データ＞フォーマットとします。

注　意　応答データは、単位を s とみなして計算、作成しますが、サフィックス部分は応答メッセージとしては送信しません。

3.4.7 DELay コマンド/ DELay? クエリ

:TIMebase:DELay <position_arg>

本コマンドで遅延時間(DELAY)を設定します。遅延時間の範囲や分解能は、その時点の時間軸設定により異なります。BPOSITION 表記も同一動作とします。

エラー処理　　:TIMebase:DELay および :TIMebase:BPOSition コマンド/クエリとも、メモリ使用種別(:ACQuire:MEMory)設定=SHORt かつ B 掃引(:TIMebase:BSWeep)設定=ON でのみ有効で、それ以外の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TIMebase:DELay <position_arg>
:TIMebase:BPOSition <position_arg>

ニーモニック　DELAY または BPOSITION
省略形　DEL または BPOS

・<position_arg> は遅延時間設定用のパラメータで、<数値プログラムデータ>フォーマットを基本とし、サフィックス部も有効とします。
・デフォルトのサフィックス乗数は×10^0、デフォルトのサフィックス単位は s とします。

例　　PRINT@DSO;":TIMEBASE:DELAY 100US"
PRINT@DSO;":TIM:BPOS 1.00E-4"

:TIMebase:DELay?

本クエリで、現在の遅延時間設定値を読み出します。BPOSition? 表記も同一動作とします。

クエリシンタックス　:TIMebase:DELay?
:TIMebase:BPOSition?

ニーモニック　DELAY? または BPOSITION?
省略形　DEL? または BPOS?

応答メッセージ　:TIMebase:DELay <position>
・「:TIMebase:DELay」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・BPOSition? クエリに対する応答の場合でも、応答メッセージのヘッダ部は DELay とします。
・<position> は<NR3 数値応答データ>フォーマットとします。

注　意　　応答データは、単位を s とみなして計算、作成しますが、サフィックス部分は応答メッセージとしては送信しません。

3.4.8 TSEParation コマンド/ TSEParation? クエリ

:TIMebase:TSEParation <tsep_div_arg>

本コマンドでトレースセパレーションを設定します。

エラー処理　　:TIMebase:TSEParation コマンド/クエリとも、メモリ使用種別(:ACQuire:MEMory)設定=SHORt かつ、B 掃引(:TIMebase:BSWeep)設定=ON でのみ有効で、それ以外の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TIMebase:TSEParation <tsep_div_arg>
ニーモニック　TSEPARATION
省略形　TSEP
<tsep_div_arg> はトレースセパレーション設定用のパラメータで、＜数値プログラムデータ＞フォーマットとします。

注　意　　設定値は、div数とみなして小数点以下2桁までの数値に丸めて解釈します。但し、サフィックス部の文字を許しません。

例　　PRINT@DSO;":TIMEBASE:TSEPARATION 4.00"
PRINT@DSO;":TIM:BPOS 4.00E-0"

:TIMebase:TSEParation?

本クエリで現在のトレースセパレーション設定値を読み出します。

クエリシンタックス　:TIMebase:TSEParation?
ニーモニック　TSEPARATION?
省略形　TSEP?

応答メッセージ　:TIMebase:TSEParation <tsep_div>
・「:TIMebase:TSEParation」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<tsep_div> は＜NR3 数値応答データ＞フォーマットとします。

注　意　　応答データの数値は div数として計算、作成しますが、サフィックス部の文字は送信しません。

3.5 CHannel 関連コマンド

:CHannel<n>:<item>…… と指定すると、チャネル<n>を対象とする各種設定が可能です。
実際の設定項目は <item> 部のヘッダにより指定されます。

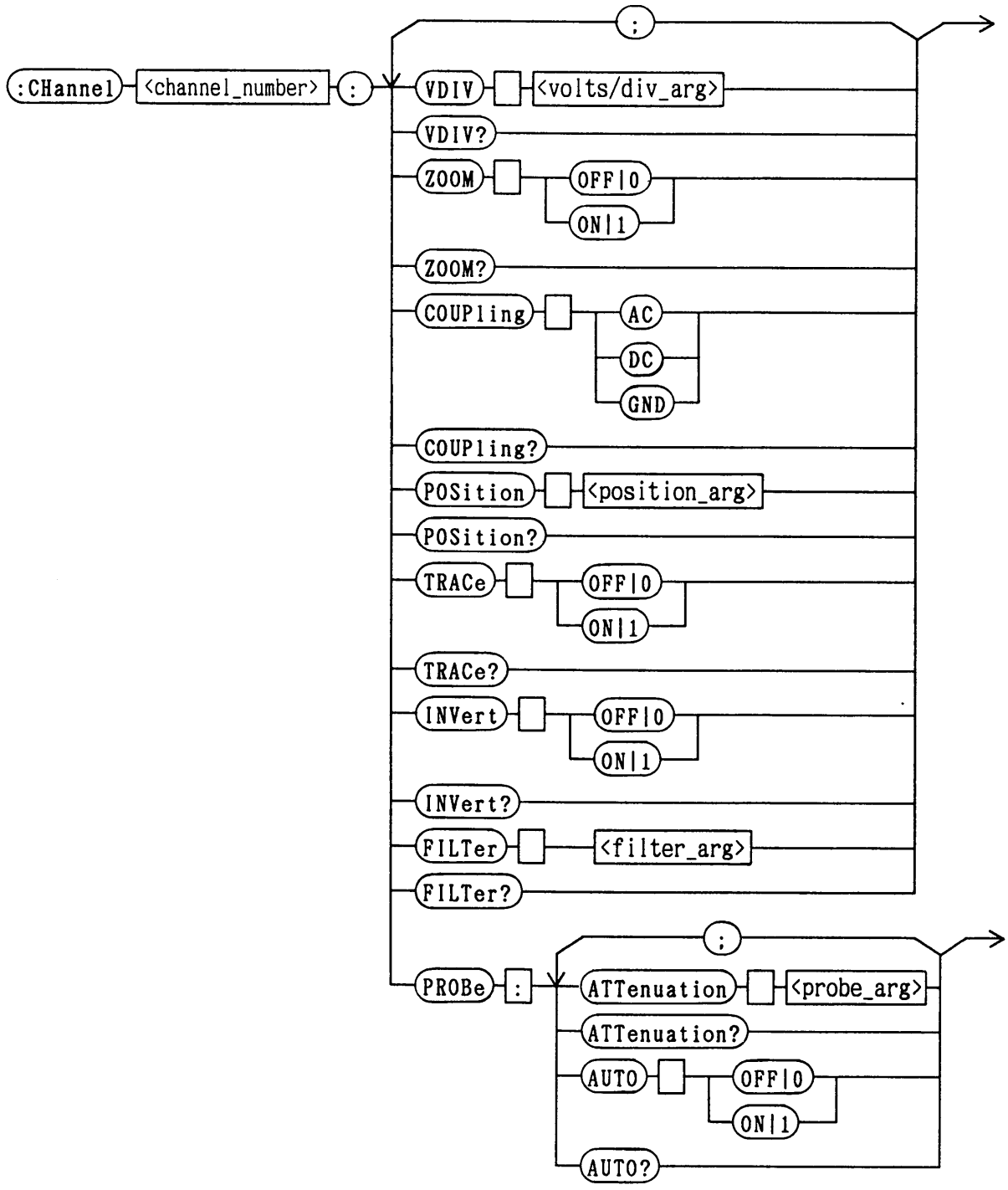
コマンドシンタックス :CHannel<n>:<item>……

ニーモニック CHANNEL1
CHANNEL2
CHANNEL3
CHANNEL4

省略形 CHAN1 または CH1
CHAN2 または CH2
CHAN3 または CH3
CHAN4 または CH4

エラー処理 チャネル番号<n>は、数値ではなく、ニーモニックを構成する ASCII 数字文字 1 文字です。従って、<n>が 5 以上の場合や、CHannel と <n> の間にスペースがある場合、<n> が 01 などと書かれている場合には、コマンドエラーとし、以下のメッセージをエラーキューに格納します。
-113,"Undefined header"

図3.5.1 CHannel関連コマンド全体像



3.5.1 VDIV コマンド/ VDIV? クエリ

:CHannel<n>:VDIV <volts/div_arg>

本コマンドでチャネル<n>の垂直軸感度(VOLTS/DIV)を設定します。

コマンドシンタックス　:CHannel<n>:VDIV <volts/div_arg>
　　ニーモニック　VDIV
　　省略形　な し
・<volts/div_arg> は垂直軸感度設定用のパラメータです。
・＜数値プログラムデータ＞フォーマットを基本とし、サフィックス部も有効とします。
・デフォルトのサフィックス乗数は×10^0、デフォルトのサフィックス単位は V とします。

エラー処理　<volts/div_arg> のサフィックス単位としては V のみ有効で、これ以外の文字を受け取った場合はコマンドエラーとし、以下のメッセージをエラーキューに格納します。
-131,"Invalid suffix"

例
```
PRINT@DSO;":CHANNEL1:VDIV 100MV"
PRINT@DSO;":CH1:VDIV 1.00E-1"
PRINT@DSO;":CHANNEL1:VDIV 120MV"
PRINT@DSO;":CH1:VDIV 1.2E-1"
```

:CHannel<n>:VDIV?

本クエリで現在の垂直軸感度設定値を読み出します。ズーム(:CHannel<n>:ZOOM)設定 ON/OFF の状態により、パラメータの丸めの方法が異なります。

クエリシンタックス　:CHannel<n>:VDIV?
　　ニーモニック　VDIV?
　　省略形　VDIV?(省略形なし)

応答メッセージ　:CHannel<n>:VDIV <volts/div>
・「:CHannel<n>:VDIV」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<volts/div> は＜NR3 数値応答データ＞フォーマットの数値とします。
・現在の垂直軸感度(VOLTS/DIV)を示します。

注　意　単位を V とみなした数値を送信しますが、サフィックス部は送信しません。

<volts/div_arg> と <volts/div> について

1)ズーム(ZOOM)機能=OFF の場合

◇設定可能な垂直軸感度は 1mV～5V の範囲の 1-2-5 ｼｰｹﾝｽのレンジです(1:1 ﾌﾟﾛｰﾌﾞ使用時)。
◇プローブ減衰比≠1の場合は、本器内部で補正します。例えば、10:1 ﾌﾟﾛｰﾌﾞを使用した場合には設定範囲が10mV～50V となります。
◇設定パラメータ<volts/div_arg>
・設定値は、まずプローブ減衰比補正を行い、丸めを行ってから解釈します。
・1-2-5 ｼｰｹﾝｽでない設定値も、それだけではエラーにはしません。
但し、＜数値プログラムデータ＞およびサフィックス部としてのフォーマットチェックの段階でコマンドエラーとなった場合を除きます。
・ズーム機能=OFF の状態での <volts/div_arg> の丸めの方法は以下の通りです。

	<volts/div_arg>≦1E-3(1mV)	→ 1mV/DIV
1E-3＜	≦2E-3(2mV)	→ 2mV/DIV
｜	｜	
2E 0＜		→ 5V/div

◇応答データ<volts/div>
・<volts/div> は＜NR3数値応答データ＞フォーマットの数値とします。
・現在の垂直軸感度(VOLTS/DIV)を示します。

2)ズーム(ZOOM)機能=ON の場合

◇設定可能な垂直軸感度は 500uV～5V の範囲です(1:1 ﾌﾟﾛｰﾌﾞ使用時)。
◇プローブ感度≠1の場合は、本器内部で補正します。
◇VDIV コマンドは、ズーム倍率とアッテネータレンジの両方の設定に影響します。
◇設定パラメータ<volts/div_arg>
・設定値は、まずプローブ減衰比補正を行い、丸めを行ってから解釈します。
・ズーム機能=ON での <volts/div_arg> は、より粗い方のいちばん近いレンジに飛びつきます。
◇応答データ<volts/div>
・<volts/div> は＜NR3 数値応答データ＞フォーマットとします。
・ズーム倍率も計算済みの現在の垂直軸感度(VOLTS/DIV)を示します。

3.5.2 ZOOM コマンド/ZOOM? クエリ

:CHannel<n>:ZOOM <mode_arg>

本コマンドでチャネル<n>のズーム ON/OFF 設定の切り換えをします。

コマンドシンタックス :CHannel<n>:ZOOM <mode_arg>
ニーモニック ZOOM
省略形 ZOOM(省略形なし)
・<mode_arg> はズーム ON/OFF 設定用のパラメータで、＜文字プログラムデータ＞フォーマットの ON、OFF または ＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
・数値を丸めて整数化した後の値が 0 の場合はOFF、0 以外の場合は ON とみなします。

例
```
PRINT@DSO;":CHANNEL1:ZOOM ON"
PRINT@DSO;":CH1:ZOOM 1"
```

:CHannel<n>:ZOOM?

本クエリで現在のズーム ON/OFF の設定状態を読み出します。

クエリシンタックス :CHannel<n>:ZOOM?
ニーモニック ZOOM?
省略形 ZOOM?(省略形なし)

応答メッセージ :CHannel<n>:ZOOM <mode>
・「:CHannel<n>:ZOOM」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の 1桁の数字とします。
・0 が OFF、1 が ON を意味します。

3.5.3 COUPling コマンド/ COUPling? クエリ

CHannel<n>:COUPling <mode_arg>

本コマンドでチャネル<n>の入力結合(AC/DC/GND)を設定します。

コマンドシンタックス　:CHannel<n>:COUPling <mode_arg>
ニーモニック　COUPLING
省略形　COUP
・<mode_arg> は入力結合(AC/DC/GND)設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・AC、DC または GND のいずれかとします。

例　PRINT@DSO;":CHANNEL1:COUPLING DC"
PRINT@DSO;":CH1:COUP dc"

:CHannel<n>:COUPling?

本クエリで現在の入力結合の設定状態を読み出します。

クエリシンタックス　:CHannel<n>:COUPling?
ニーモニック　COUPLING?
省略形　COUP?

応答メッセージ　:CHannel<n>:COUPling <mode>
・「:CHannel<n>:COUPling」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は＜文字応答データ＞フォーマットのニーモニックで、現在の入力結合を示します。
・AC、DC または GND のいずれかとなります。

3.5.4 POSition コマンド/ POSition? クエリ

:CHannel<n>:POSition <position_arg>

本コマンドでチャネル<n>の垂直位置(POSITION)を設定します。設定範囲は垂直軸感度の-12.0～+12.0div の範囲に相当する電圧値です。

コマンドシンタックス　:CHannel<n>:POSition <position_arg>
　　ニーモニック　POSITION
　　省略形　POS

・<position_arg> は垂直位置設定用のパラメータで、＜数値プログラムデータ＞フォーマットを基本とし、サフィックス部も有効とします。
・デフォルトのサフィックス乗数は×10^0、デフォルトのサフィックス単位は V とします。

エラー処理　<volts/div_arg> のサフィックス単位としては V のみ有効で、これ以外の文字を受け取った場合はコマンドエラーとし、以下のメッセージをエラーキューに格納します。
-131,"Invalid suffix"

例
```
PRINT@DSO;":CHANNEL1:POSITION 1.2V"
PRINT@DSO;":CH1:POS 12E-1"
```

:CHannel<n>:POSition?

本クエリで現在の垂直位置(POSITION)設定値を読み出します。

クエリシンタックス　:CHannel<n>:POSition?
　　ニーモニック　POSITION?
　　省略形　POS?

応答メッセージ　:CHannel<n>:POSition <position>

・「:CHannel<n>:POSition」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<position> は＜NR3 数値応答データ＞フォーマットの数値で、現在の垂直位置を示します。

3.5.5 TRACe コマンド/ TRACe? クエリ

:CHannel<n>:TRACe <mode_arg>

本コマンドでチャネル<n>のトレース表示(ON/OFF)を設定します。

コマンドシンタックス　:CHannel<n>:TRACe <mode_arg>
　　ニーモニック　TRACE
　　省略形　TRAC

・<mode_arg> はON/OFF 設定用のパラメータで、文字プログラムデータ＞フォーマットの ON、OFF または＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
・数値を丸めて整数化した後の値が 0 の場合は OFF、0 以外の場合は ON とみなします。

エラー処理

・B 掃引(:TIMebase:BSWeep)設定=ON の状態で、CH3 または CH4 のトレースを ON に設定しようとした場合には実行エラーとします。
・また、アクイジションチャネル設定=CH1&2 となるようなメモリ長に指定されている状態で、CH3 または CH4 のトレースを ON に設定しようとした場合も同様です。
・同様に、アクイジションチャネル設定=CH1_only となるようなメモリ長に指定されている状態で、CH1 以外のトレースを ON に設定しようとした場合にも実行エラーとします。
・いずれの場合も、以下のメッセージをエラーキューに格納します。
　-221,"Setting conflict"

例

```
PRINT@DSO;":CHANNEL1:TRACE ON"
PRINT@DSO;":CH1:TRAC 1"
```

:CHannel<n>:TRACe?

本クエリで、現在のトレース表示(ON/OFF)の設定を読み出します。

クエリシンタックス　:CHannel<n>:TRACe?
　　ニーモニック　TRACE?
　　省略形　TRAC?

応答メッセージ　:CHannel<n>:TRACe <mode>

・「:CHannel<n>:TRACe」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットの 0 または 1 の 1 桁の数字で現在のトレース設定を示します。
・0 が OFF、1 が ON を意味します。

備　考

前出の実行エラーを引き起こすような組み合わせのチャネルに対する TRACe? クエリを受信した場合の応答は 0(OFF を意味する) とします。

3.5.6 INVert コマンド/ INVert? クエリ

:CHannel<n>:INVert <mode_arg>

本コマンドでチャネル<n>のインバート(ON/OFF)を設定します。

コマンドシンタックス　:CHannel<n>:INVert <mode_arg>
ニーモニック　INVERT
省略形　INV

・<mode_arg>は ON/OFF 設定用のパラメータで、＜文字プログラムデータ＞フォーマットの ON、OFF または ＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
・数値を丸めて整数化した後の値が 0 の場合は OFF、0 以外の場合は ON とみなします。

エラー処理

・B 掃引(:TIMebase:BSWeep)設定=ON の状態で、CH3 または CH4 のインバート機能を ON に設定しようとした場合には実行エラーとします。
・また、アクイジションチャネル設定=CH1&2 となるようなメモリ長に指定されている状態で CH3 または CH4 のインバート機能を ON に設定しようとした場合も実行エラーとします。
・同様に、アクイジションチャネル設定=CH1_only となるようなメモリ長に指定されている状態で、CH1 以外のインバート機能を ON に設定しようとした場合にも実行エラーとします。
・いずれの場合も、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例

```
PRINT@DSO;":CHANNEL1:INVERT ON"
PRINT@DSO;":CH1:INV 1"
```

:CHannel<n>:INVert?

本クエリで現在のインバート(ON/OFF)の設定を読み出します。

クエリシンタックス　:CHannel<n>:INVert?
ニーモニック　INVERT?
省略形　INV?

応答メッセージ　:CHannel<n>:INVert <mode>

・「:CHannel<n>:INVert」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットの 0 または 1 の 1 桁の数字とします。
・0 が OFF、1 が ON を意味します。

備　考　前出の実行エラーを引き起こすような組み合わせのチャネルに対する INVert? クエリを受信した場合の応答は 0 (OFF を意味する)とします。

3.5.7 FILTer コマンド/ FILTer? クエリ　（DS−9121 のみ）

:CHannel<n>:FILTer <filter_arg>

本コマンドでチャネル<n>のローパスフィルタの周波数を設定します。設定可能な周波数は 100Hz、1kHz または 10kHz です。OFF にする場合は 0 に設定します。本コマンドは DS-9121 のみ有効です。

コマンドシンタックス　:CHannel<n>:FILTer <filter_arg>

ニーモニック　FILTER

省略形　FILT

・<filter_arg> はフィルター周波数設定用のパラメータで、＜数値プログラムデータ＞フォーマットを基本とし、サフィックス部も有効とします。
・デフォルトのサフィックス乗数は×10⁰、デフォルトのサフィックス単位は Hz とします。
・<filter_arg> が設定可能な値と一致しない場合、より高い方のいちばん近い周波数に飛びつきます。但し、10kHz をこえる周波数を指定された場合は 10kHz に設定します。また、加数部の数値を丸めた結果が 0 の場合は、OFF とみなします。

エラー処理

・<filter_arg> のサフィックス単位としては Hz のみ有効で、これ以外の文字を受け取った場合はコマンドエラーとし、以下のメッセージをエラーキューに格納します。
-131,"Invalid suffix"
・B 掃引(:TIMebase:BSWeep)設定=ON の状態で、CH3 または CH4 のフィルタを OFF 以外に設定しようとした場合には実行エラーとします。
・また、アクイジションチャネル設定=CH1&2 となるようなメモリ長に指定されている状態で、CH3 または CH4 のフィルタを OFF 以外に設定しようとした場合も実行エラーとします。
・同様に、アクイジションチャネル設定=CH1_only となるようなメモリ長に指定されている状態で、CH1 以外のフィルタを OFF 以外に設定しようとした場合にも実行エラーとします。
・いずれの場合も、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"
・DS-9122 で本コマンドを受信した場合はコマンドエラーとし、以下のメッセージをエラーキューに格納します。
-113 "Undefined header"

例

```
PRINT@DSO;":CHANNEL1:FILTER 100KHZ"
PRINT@DSO;":CH1:FILT 1E5"
```

:CHannel<n>:FILTer?

本クエリで現在のフィルター周波数の設定値を読み出します。

クエリシンタックス	:CHannel<n>:FILTer?
ニーモニック	FILTER?
省略形	FILT?
応答メッセージ	:CHannel<n>:FILTer <filter> ・「:CHannel<n>:FILTer」部分は :SYSTem:HEADer 設定＝OFF の場合には省略します。 ・<filter> は＜NR3 数値応答データ＞フォーマットの数値で、現在のフィルタ周波数を示します。 ・単位を Hz とみなした数値を送信しますが、サフィックス部は送信しません。 ・OFF の場合は 0E+0 を返します。
備　考	前出の実行エラーを引き起こすような組み合わせのチャネルに対する FILTer? クエリを受信した場合の応答は 0E+0(OFF を意味する)とします。

3.5.8 PROBe コマンド/ PROBe? クエリ

:CHannel<n>:PROBe:<item>……　と指定すると、チャネル<n>を対象とするプローブ関連の各種設定が可能です。実際の設定項目は <item> 部のヘッダにより指定されます。

コマンドシンタックス　:CHannel<n>:PROBe:<item>……
　　ニーモニック　PROBE
　　省略形　　PROB

3.5.8.1 ATTenuation コマンド/ ATTenuation? クエリ

:CHannel<n>:PROBe:ATTenuation <probe_arg>

本コマンドでチャネル<n>のプローブの減衰比を設定します。設定範囲は 0.1～1000.0 です。

コマンドシンタックス　:CHannel<n>:PROBe:ATTenuation <probe_arg>
　　ニーモニック　ATTENUATION
　　省略形　　ATT
　　・<probe_arg> はプローブ減衰比設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値です。
　　・設定値は、小数点以下 1 桁までの実数値に丸めを行ってから解釈します。設定値が設定可能範囲をこえる場合は、設定可能範囲内の上限あるいは下限に飛びつきます。

例　　PRINT@DSO;":CHANNEL1:PROBE:ATTENUATION 10"
　　PRINT@DSO;":CH1:PROB:ATT 1.0E+1"

注　意　　本コマンドで設定した値は、プローブを MANUAL に設定したときに使用します。

:CHannel<n>:PROBe:ATTenuation?

本クエリで現在のプローブ減衰比を読み出します。

クエリシンタックス　:CHannel<n>:PROBe:ATTenuation?
　　ニーモニック　ATTENUATION?
　　省略形　　ATT?

応答メッセージ　:CHannel<n>:PROBe:ATTenuation <probe>
　　・「:CHannel<n>:PROBe:ATTenuation」部分は :SYSTem:HEADer設定＝OFF の場合には省略します。
　　・<probe> は＜NR3 数値応答データ＞フォーマットの数値で、現在のプローブ減衰比を示します。

3.5.8.2 AUTO コマンド/ AUTO? クエリ

:CHannel<n>:PROBe:AUTO

本コマンドでチャネル<n>のプローブの AUTO/MANUAL の設定をします。

コマンドシンタックス :CHannel<n>:PROBe:AUTO <mode_arg>
ニーモニック AUTO
省略形 AUTO

・<mode_arg> は ON/OFF 設定用のパラメータで＜文字プログラムデータ＞フォーマットの ON、OFF または＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
・数値を丸めて整数化した後の値が 0 の場合は OFF(MANUAL)、0 以外の場合は ON(AUTO) とみなします。

例
```
PRINT@DSO;":CHANNEL1:PROBE:AUTO"
PRINT@DSO;":CH1:PROB:auto"
```

:CHannel<n>:PROBe:AUTO?

本クエリでチャネル<n>のプローブの AUTO/MANUAL の設定を読み出します。

クエリシンタックス :CHannel<n>:PROBe:AUTO?
ニーモニック AUTO?
省略形 AUTO?

応答メッセージ :CHannel<n>:PROBe:AUTO <mode>

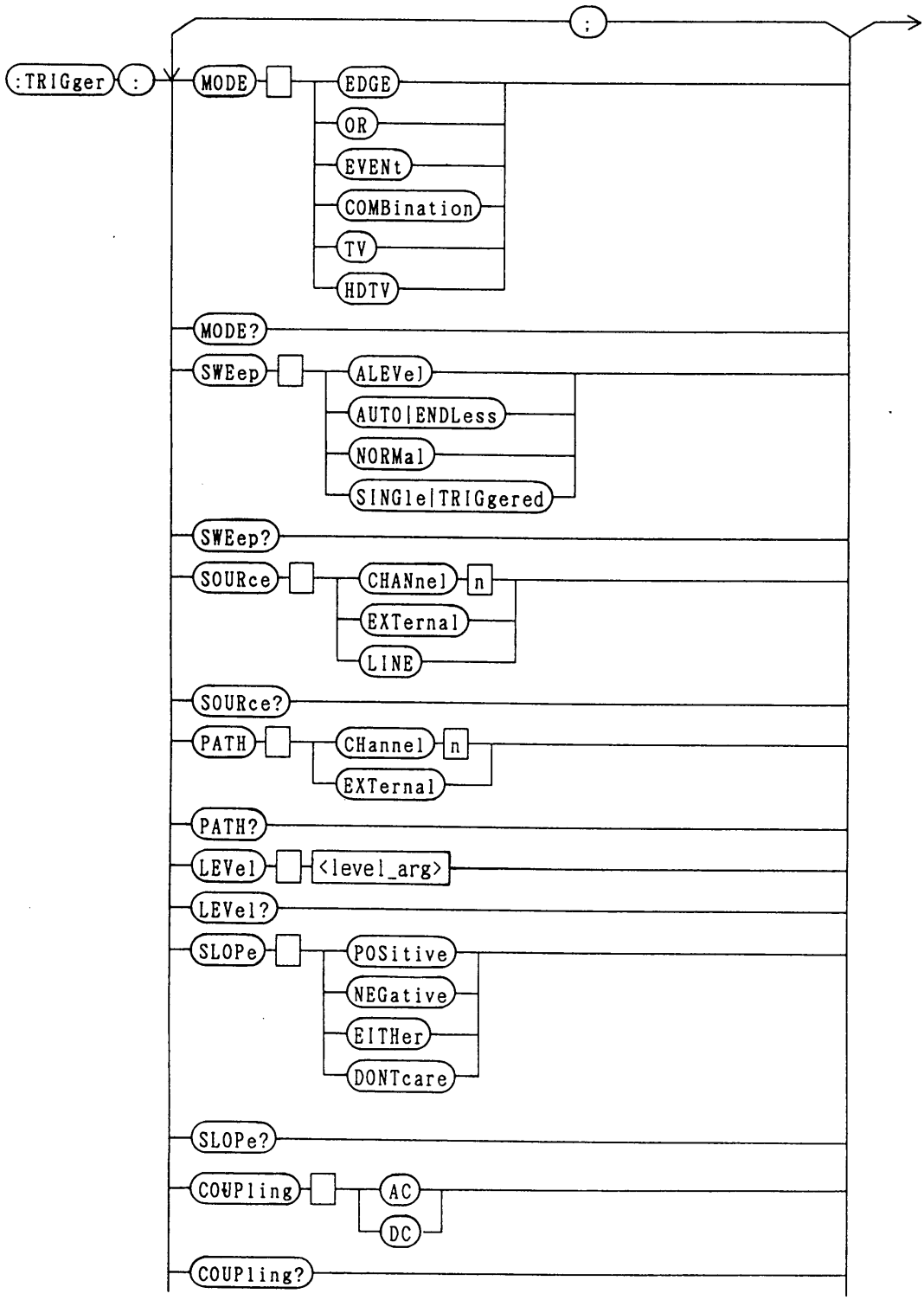
・「:CHannel<n>:PROBe:AUTO」部分は :SYSTem:HEADer 設定＝OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の1 桁の数字とします。
・0 が OFF(MANUAL)、1 が ON(AUTO) を意味します。

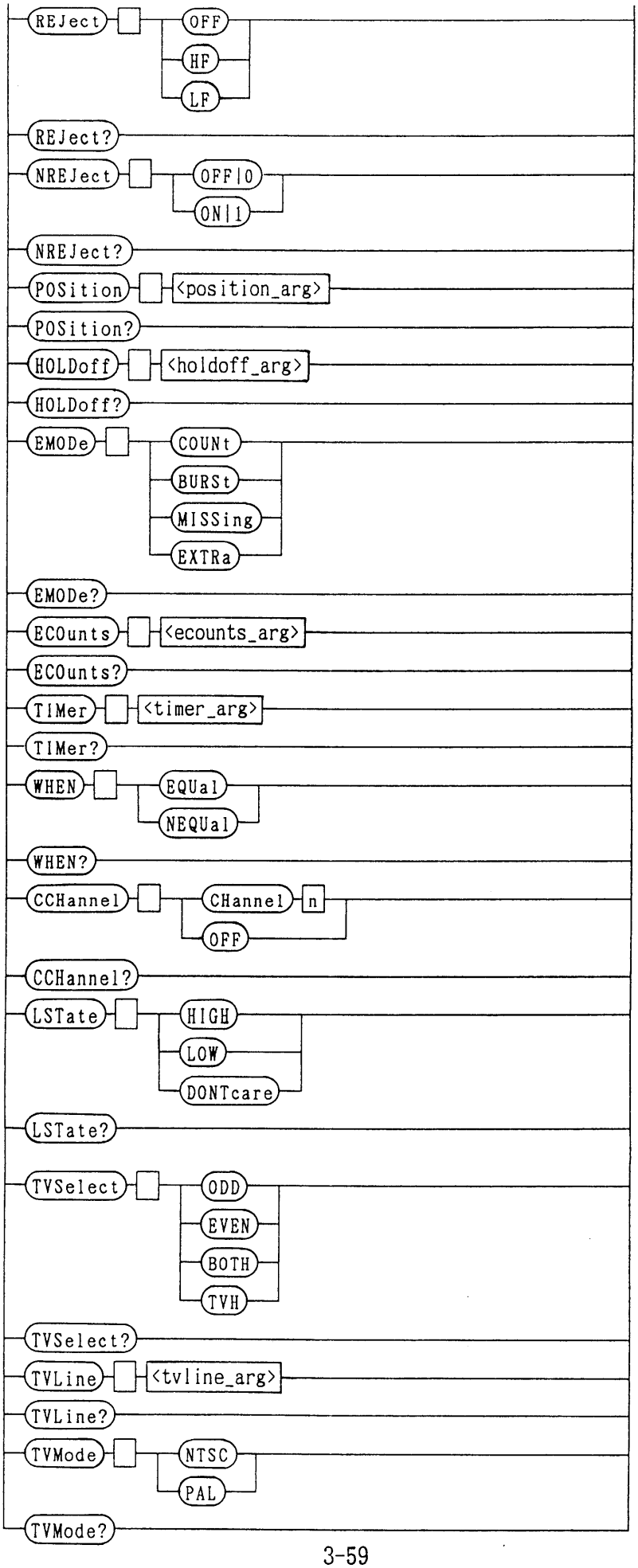
3.6 TRIGger 関連コマンド

:TRIGger:<item>…… と指定することにより、トリガ機能関連の各種設定が可能です。実際の設定項目は <item> 部のヘッダにより指定されます。
尚、イベントトリガのアーム信号関連の項目は ARM 親コマンドとして独立させます。

コマンドシンタックス :TRIGger:<item>……
ニーモニック TRIGGER
省略形 TRIG

図3.6.1 TRIGger 関連コマンド全体像




REJect
OFF
HF
LF
REJect?
NREJect
OFF|0
ON|1
NREJect?
POSition
<position_arg>
POSition?
HOLDoff
<holdoff_arg>
HOLDoff?
EMODe
COUNt
BURSt
MISSing
EXTRa
EMODe?
ECOunts
<ecounts_arg>
ECOunts?
TIMer
<timer_arg>
TIMer?
WHEN
EQUal
NEQUal
WHEN?
CCHannel
CHannel
n
OFF
CCHannel?
LSTate
HIGH
LOW
DONTcare
LSTate?
TVSelect
ODD
EVEN
BOTH
TVH
TVSelect?
TVLine
<tvline_arg>
TVLine?
TVMode
NTSC
PAL
TVMode?

3.6.1 MODE コマンド/ MODE? クエリ

:TRIGger:MODE <trig_mode_arg>

本コマンドでトリガモードを設定します。トリガモードの設定に応じて :TRIGger 親コマンド下の各コマンドのエラー判定と :ARM 親コマンド下の各コマンドのエラー判定が行なわれます。
:TRIGger 親コマンドまたは :ARM 親コマンド下のサブコマンドを使用する場合には、最初に目的のトリガモードを設定して下さい。

コマンドシンタックス　:TRIGger:MODE <trig_mode_arg>
ニーモニック　MODE
省略形　MODE(省略形なし)
・<trig_mode_arg> はトリガモード設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・EDGE、OR、EVENt、COMBination、TV、HDTV の何れかとします。

エラー処理　DS-9121 で TV/HDTV トリガオプションが実装されていない場合、TV または HDTV 指定はコマンドエラーとし、以下のメッセージをエラーキューに格納します。
-141,"Invalid character data"

例
```
PRINT@DSO;":TRIGGER:MODE EDGE"
PRINT@DSO;":TRIG:MODE EDGE"
```

:TRIGger:MODE?

本クエリで、現在のトリガモードの設定を読み出します。

クエリシンタックス　:TRIGger:MODE?
ニーモニック　MODE?
省略形　MODE?(省略形なし)

応答メッセージ　:TRIGger:MODE <trig_mode>
・「:TRIGger:MODE」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<trig_mode> は＜文字応答データ＞フォーマットのニーモニックです。
・EDGE、OR、EVENt、COMBination、TV、HDTV のいずれかとします。

3.6.2 SWEep コマンド/ SWEep? クエリ

:TRIGger:SWEep <sweep_arg>

本コマンドで掃引方式の設定をします。

コマンドシンタックス　:TRIGger:SWEep <sweep_arg>
　　ニーモニック　SWEEP
　　省略形　SWE

・<sweep_arg>は掃引方式設定用のパラメータで＜文字プログラムデータ＞フォーマットのニーモニックです。
・ALEVel、AUTO、NORMal、SINGle、TRIGgered、ENDLess のいずれかとします。
・AUTO と ENDLess は同義と解釈します。
　同様に、SINGle と TRIGgered も同義と解釈します。

エラー処理

・メモリ使用種別(:ACQuire:MEMory)設定=LONG または ADVance の状態で SINGle(またはTRIGggered) 以外の掃引方式を指定された場合、実行エラーとします。
・メモリ使用種別(:ACQuire:MEMory)設定=ROLL の状態で ALEVel または NORMAL を掃引方式として指定された場合も実行エラーとします。
・ALEVel 設定はトリガモード(:TRIGger:MODE)設定=EDGE の場合のみ有効で、それ以外では、実行エラーとします。
・いずれの場合も、以下のメッセージをエラーキューに格納します。
　-221,"Setting conflict"

例

```
PRINT@DSO;":TRIGGER:SWEEP NORMAL"
PRINT@DSO;":TRIG:SWE NORM"
```

:TRIGger:SWEep?

本クエリで、現在の掃引方式の設定を読み出します。

クエリシンタックス　:TRIGger:SWEep?
　　ニーモニック　SWEEP?
　　省略形　SWE?

応答メッセージ　:TRIGger:SWEep <sweep>

・「:TRIGger:SWEep」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<sweep> は＜文字応答データ＞フォーマットのニーモニックです。
・ALEVel、AUTO、NORMal、SINGle (ROLL モード以外) または TRIGgered、ENDLess (ROLL モード) のいずれかとします。

3.6.3 SOURce コマンド/ SOURce? クエリ

:TRIGger:SOURce <source_arg>

本コマンドでトリガ信号源(イベントトリガモードでは、イベント信号源)を設定します。
また、:TRIGger:SOURce <source_arg> コマンドは、現在の :TRIGger:PATH 設定を <source_arg> 指定と一致させます。

エラー処理　SOURce コマンド/クエリ共、トリガモード(:TRIGger:MODE)設定が EDGE、EVENT、TV、HDTV の何れでもない場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TRIGger:SOURce <source_arg>
ニーモニック　SOURCE
省略形　SOUR

・<source_arg> はトリガ信号源設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・CHannel1、CHannel2、CHannel3、CHannel4、EXTernal、LINE のいずれかとします。

エラー処理　EXTernal または LINE 指定はトリガモード(:TRIGger:MODE)設定＝EDGEの場合のみ有効で、それ以外では実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例
```
PRINT@DSO;":TRIGGER:SOURCE CHANNEL1"
PRINT@DSO;":TRIG:SOUR CH2"
```

:TRIGger:SOURce?

本クエリで現在のトリガ信号源の設定を読み出します。

クエリシンタックス　:TRIGger:SOURce?
ニーモニック　SOURCE?
省略形　SOUR?

応答メッセージ　:TRIGger:SOURce <source>

・「:TRIGger:SOURce」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<source> は＜文字応答データ＞フォーマットのニーモニックです。
・CHANnel1、CHANnel2、CHANnel3、CHANnel4、EXTernal、LINE のいずれかです。

3.6.4 PATH コマンド/PATH? クエリ

:TRIGger:PATH <path_arg>

本コマンドで SLOPe、COUPling、REJect、NREJect、LEVel など、チャネル毎に設定可能な各種トリガ設定に関するサブコマンドの対象チャネルを選択します。
トリガモード(:TRIGger:MODE)設定が OR または COMBination の場合にはエディットチャネルの設定に該当します。これ以外のトリガモード(:TRIGger:MODE)指定では該当するローカル操作はありません。ただし、:TRIGger:SOURce 設定が行われた場合、:TRIGger:PATH 設定をこれに一致させるものとします。

コマンドシンタックス　:TRIGger:PATH <path_arg>
　　ニーモニック　PATH
　　省略形　PATH(省略形なし)
　　・<path_arg> はトリガパス設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
　　・CHannel1、CHannel2、CHannel3、CHannel4、EXTernal、LINE のいずれかとします。

エラー処理
　　・LINE 指定はトリガモード(:TRIGger:MODE)設定=EDGE の場合のみ有効で、それ以外では実行エラーとします。
　　・EXTernal 指定はトリガモード(:TRIGger:MODE)設定が EDGE または COMBination の場合のみ有効で、それ以外では実行エラーとします。
　　・いずれの場合も、以下のメッセージをエラーキューに格納します。
　　-221,"Setting conflict"

例
```
PRINT@DSO;":TRIGGER:PATH CHANNEL1"
PRINT@DSO;":TRIG:PATH CH2"
```

:TRIGger:PATH?

本クエリで、現在の :TRIGger:PATH 設定を読み出します。

クエリシンタックス　:TRIGger:PATH?
　　ニーモニック　PATH?
　　省略形　PATH?(省略形なし)

応答メッセージ　:TRIGger:PATH <path>
　　・「:TRIGger:PATH」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<path> は＜文字応答データ＞フォーマットのニーモニックです。
　　・CHANnel1、CHANnel2、CHANnel3、CHANnel4、EXTernal、LINE のいずれかです。

3.6.5 LEVel コマンド/ LEVel? クエリ

:TRIGger:LEVel <level_arg>

本コマンドで :TRIGger:PATH 指定されたチャネルのトリガレベルを設定します。

エラー処理　　:TRIGger:PATH 指定=EXTernal に設定されている場合は LEVel サブコマンドによる設定は無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TRIGger:LEVel <level_arg>
ニーモニック　LEVEL
省略形　　LEV

・<level_arg> はトリガレベル設定パラメータで＜数値プログラムデータ＞フォーマットを基本とし、サフィックス部も有効とします。
・デフォルトのサフィックス乗数は×10^0、デフォルトのサフィックス単位は V(PATH≠LINE)、PCT(PATH=LINE) とします。
・設定可能範囲は、そのときの垂直軸感度(VOLTS/DIV)設定の ±16div に相当する電圧値(PATH≠LINE)、または ±100%(PATH=LINE)です。

エラー処理　　<level_arg> のサフィックス単位としては TRIGGER PATH≠LINE では V のみ有効、TRIGGER PATH=LINE では PCT のみ有効で、これ以外の文字を受け取った場合はコマンドエラーとし、以下のメッセージをエラーキューに格納します。
-131,"Invalid suffix"

例　　PRINT@DSO;":TRIGGER:LEVEL 500mV"
PRINT@DSO;":TRIG:LEV 5E-1"

:TRIGger:LEVel?

本クエリで、現在のトリガレベル設定を読み出します。

クエリシンタックス　:TRIGger:LEVel?
ニーモニック　LEVEL?
省略形　　LEV?

応答メッセージ　　:TRIGger:LEVel <level>
・「:TRIGger:LEVel」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<level> は＜NR3 数値応答データ＞フォーマットとします。
・TRIGGER PATH設定＝EXTernal の場合には以下の応答データとします。
1.5E+0

3.6.6 SLOPe コマンド/ SLOPe? クエリ

:TRIGger:SLOPe <slope_arg>

本コマンドで :TRIGger:PATH 指定された同期信号源を対象とするトリガスロープを設定します。

エラー処理　トリガモード(:TRIGger:MODE)設定=COMBination に設定されている場合は、:TRIGger:PATH 指定と :TRIGger:CCHannel 指定とが一致している場合のみ SLOPe サブコマンドによる設定が有効で、それ以外では実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TRIGger:SLOPe <slope_arg>
ニーモニック　SLOPE
省略形　SLOP
・<slope_arg> はトリガスロープ設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・POSitive、NEGative、EITHer、DONTcare のいずれかとします。
・ローカル操作のメニュー表示では、POSitive は ＋、NEGative －、EITHer は ±、DONTcare は X に相当します。

エラー処理
・DONTcare 指定はトリガモード(:TRIGger:MODE)設定=OR の場合のみ有効で、それ以外では実行エラーとします。
・:TRIGger:PATH 指定=EXTernal の場合には EITHer 指定は無効で、実行エラーとします。
・いずれの場合も、以下のメッセージをエラーキューに格納します。
-221、"Setting conflict"

例
```
PRINT@DSO;":TRIGGER:SLOPE POSITIVE"
PRINT@DSO;":TRIG:SLOP POS"
```

:TRIGger:SLOPe?

本クエリで、現在のトリガスロープ設定を読み出します。

クエリシンタックス　:TRIGger:SLOPe?
ニーモニック　SLOPE?
省略形　SLOP?

応答メッセージ　:TRIGger:SLOPe <slope>
・「:TRIGger:SLOPe」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<slope> は＜文字応答データ＞フォーマットのニーモニックです。
・POSitive、NEGative、EITHer、DONTcare のいずれかです。

3.6.7 COUPling コマンド/ COUPling? クエリ

:TRIGger:COUPling <coupling_arg>

本コマンドで :TRIGger:PATH 指定されたチャネルを対象とする同期結合の設定をします。

エラー処理
- トリガモード(:TRIGger:MODE)設定が TV または HDTV の場合は COUPLING サブコマンドによる設定は無効で、実行エラーとします。
- :TRIGger:PATH 指定が EXTernal または LINE の場合は COUPLING サブコマンドによる設定は無効で、実行エラーとします。
- いずれの場合も以下のメッセージをエラーキューに格納します。
  -221,"Setting conflict"

コマンドシンタックス :TRIGger:COUPling <coupling_arg>
ニーモニック COUPLING
省略形 COUP
- <coupling_arg> は同期結合設定パラメータで<文字プログラムデータ>フォーマットのニーモニックです。
- AC あるいは DC のいずれかとします。

例
```
PRINT@DSO;":TRIGGER:COUPLING DC"
PRINT@DSO;":TRIG:COUP dc"
```

:TRIGger:COUPling?

本クエリで、現在の同期結合設定を読み出します。

クエリシンタックス :TRIGger:COUPling?
ニーモニック COUPLING?
省略形 COUP?

応答メッセージ :TRIGger:COUPling <coupling>
- 「:TRIGger:COUPling」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
- <coupling> は<文字応答データ>フォーマットのニーモニックです。
- AC または DC のいずれかとします。

3.6.8 REJect コマンド/ REJect? クエリ

:TRIGger:REJect <reject_arg>

本コマンドで :TRIGger:PATH 指定されたチャネルの HF/LF リジェクションを設定します。

エラー処理
- トリガモード(:TRIGger:MODE)設定が TV または HDTV の場合には REJect サブコマンドは無効で、実行エラーとします。
- :TRIGger:PATH 指定が EXTernal または LINE の場合も REJECT サブコマンドは無効で、実行エラーとします。
- いずれの場合も、以下のメッセージをエラーキューに格納します。
  -221,"Setting conflict"

コマンドシンタックス :TRIGger:REJect <reject_arg>
ニーモニック REJECT
省略形 REJ
- <reject_arg> は HF/LF リジェクション設定パラメータで<文字プログラムデータ>フォーマットのニーモニックです。
- OFF、HF、LF のいずれかとします。

例
```
PRINT@DSO;":TRIGGER:REJECT HF"
PRINT@DSO;":TRIG:REJ HF"
```

:TRIGger:REJect?

本クエリで現在の HF/LF リジェクション設定を読み出します。

クエリシンタックス :TRIGger:REJect?
ニーモニック REJECT?
省略形 REJ?

応答メッセージ :TRIGger:REJect <reject>
- 「:TRIGger:REJect」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
- <reject> は<文字応答データ>フォーマットのニーモニックです。
- OFF、HF、LF のいずれかとします。

3.6.9 NREJect コマンド/ NREJect? クエリ

:TRIGger:NREJect <mode_arg>

本コマンドで :TRIGger:PATH 指定されたチャネルのノイズリジェクション ON/OFF を設定します。

エラー処理
- トリガモード(:TRIGger:MODE)設定が TV または HDTV の場合は NREJect サブコマンドによる設定は無効で、実行エラーとします。
- :TRIGger:PATH 指定が EXTernal または LINE の場合は NREJect サブコマンドによる設定は無効で、実行エラーとします。
- いずれの場合も、以下のメッセージをエラーキューに格納します。
  -221,"Setting conflict"

コマンドシンタックス :TRIGger:NREJect <mode_arg>
ニーモニック NREJECT
省略形 NREJ

<mode_arg> はノイズリジェクション設定パラメータで＜文字プログラムデータ＞フォーマットの ON または OFF、＜数値プログラムデータ＞フォーマットの数値のいずれかとします。

例
```
PRINT@DSO;":TRIGGER:NREJECT ON"
PRINT@DSO;":TRIG:NREJ 1"
```

:TRIGger:NREJect?

本クエリで、現在のノイズリジェクション設定を読み出します。

クエリシンタックス :TRIGger:NREJect?
ニーモニック NREJECT?
省略形 NREJ?

応答メッセージ :TRIGger:NREJect <mode>
- 「:TRIGger:NREJect」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
- <mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の 1桁の数字とします。
- 0 が OFF、1 が ON を意味します。

3.6.10 POSition コマンド/ POSition? クエリ

:TRIGger:POSition <position_arg>

本コマンドでトリガポジションを設定します。

コマンドシンタックス　:TRIGger:POSition <position_arg>
　　ニーモニック　POSITION
　　省略形　POS
　　<position_arg> はトリガポジション設定パラメータで、＜数値プログラムデータ＞フォーマットとし、ローカルメニュー表示 n/10 の n を設定するものです。

例　PRINT@DSO;":TRIGGER:POSITION 5"
　　PRINT@DSO;":TRIG:POS 5E+0"

:TRIGger:POSition?

本クエリで、現在のトリガポジションの設定を読み出します。

クエリシンタックス　:TRIGger:POSition?
　　ニーモニック　POSITION?
　　省略形　POS?

応答メッセージ　:TRIGger:POSition <position>
・「:TRIGger:POSition」部分は :SYSTem:HEADer 設定＝OFF の場合には省略します。
・<position> は＜NR1 数値応答データ＞フォーマットの 1桁の数値です。

3.6.11 HOLDoff コマンド/ HOLDoff? クエリ

:TRIGger:HOLDoff <holdoff_arg>

本コマンドでホールドオフ時間を設定します。

コマンドシンタックス　:TRIGger:HOLDoff <holdoff_arg>
　　ニーモニック　HOLDOFF
　　省略形　HOLD
　　・<holdoff_arg> はホールドオフ時間設定パラメータで＜数値プログラムデータ＞フォーマットを基本とし、サフィックス部も有効とします。
　　・デフォルトのサフィックス乗数は×10^0、デフォルトのサフィックス単位は s とします。

例　PRINT@DSO;":TRIGGER:HOLDOFF 50ms"
　　PRINT@DSO;":TRIG:HOLD 5E-2"

:TRIGger:HOLDoff?

本クエリで、現在のホールドオフ時間の設定を読み出します。

クエリシンタックス　:TRIGger:HOLDoff?
　　ニーモニック　HOLDOFF?
　　省略形　HOLD?

応答メッセージ　:TRIGger:HOLDoff <holdoff>
　　・「:TRIGger:HOLDoff」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<holdoff> は＜NR3 数値応答データ＞フォーマットの数値とします。
　　・単位を s とみなして数値を計算しますが、サフィックス部の文字は送信しません。

3.6.12 イベントトリガ専用コマンド

トリガモード(:TRIGger:MODE)設定=EVENT の場合にのみ使用するサブコマンド群を規定します。
但し、アーム条件に関する設定は :ARM 親コマンド下に独立させるため、本節には記載しません。

エラー処理　　トリガモード(:TRIGger:MODE)設定≠EVENTの場合は、EMODe、ECOunts、TIMer の 3 種のコマンド/クエリは無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

3.6.12.1 EMODe コマンド/ EMODe? クエリ

:TRIGger:EMODe <event_mode_arg>

本コマンドでイベントモードを設定します。

コマンドシンタックス　:TRIGger:EMODe <event_mode_arg>
ニーモニック　EMODE
省略形　EMOD
・<event_mode_arg> はイベントモード設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・COUNt、BURSt、MISSing、EXTRa のいずれかとします。

例　PRINT@DSO;":TRIGGER:EMODE COUNT"
PRINT@DSO;":TRIG:EMOD COUN"

:TRIGger:EMODe?

本クエリで、現在のイベントモードの設定を読み出します。

クエリシンタックス　:TRIGger:EMODe?
ニーモニック　EMODE?
省略形　EMOD?

応答メッセージ　:TRIGger:EMODe <event_mode>
・「:TRIGger:EMODe」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<event_mode> は＜文字応答データ＞フォーマットのニーモニックです。
・COUNt、BURSt、MISSing、EXTRa のいずれかです。

3.6.12.2 ECOunts コマンド/ ECOunts? クエリ

:TRIGger:ECOunts <event_counts_arg>

本コマンドで :TRIGger:EMODe 設定=COUNT の場合のイベントカウント値の設定をします。

エラー処理　　イベントモード(:TRIGger:EMODe)設定≠COUNT の場合には、ECOuntsコマンド/クエリは無効(TIMerコマンド/クエリを使用すべき)で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TRIGger:ECOunts <event_counts_arg>
ニーモニック　ECOUNTS
省略形　ECO
<event_counts_arg> はイベントカウント値設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値とします。

エラー処理　　<event_counts_arg> として負の数を設定しようとした場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-222,"Data out of range"

例　　PRINT@DSO;":TRIGGER:ECOUNTS COUNT"
PRINT@DSO;":TRIG:ECO COUN"

:TRIGger:ECOunts?

本クエリで、現在のイベントカウント値の設定を読み出します。

クエリシンタックス　:TRIGger:ECOunts?
ニーモニック　ECOUNTS?
省略形　ECO?

応答メッセージ　:TRIGger:ECOunts <event_counts>
・「:TRIGger:ECOunts」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<event_counts> は＜NR1 数値応答データ＞フォーマットの整数です。

3.6.12.3 TIMer コマンド/ TIMer? クエリ

:TRIGger:TIMer <event_timer_arg>

本コマンドで :TRIGger:EMODe設定≠COUNT の場合のイベントバリューを設定します。

エラー処理　　イベントモード(:TRIGger:EMODe)設定=COUNT の場合は、:TRIGger:TIMer コマンド/クエリは無効(ECOunts を使用すべき)で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TRIGger:TIMer <event_timer_arg>
ニーモニック　TIMER
省略形　TIM

・<event_timer_arg> はイベントバリュー設定用のパラメータで ＜数値プログラムデータ＞フォーマットを基本とし、サフィックス部も有効とします。
・デフォルトのサフィックス乗数は×10^0、デフォルトのサフィックス単位は s とします。

エラー処理　　<event_timer_arg> として負の数を設定しようとした場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-222,"Data out of range"

例
```
PRINT@DSO;":TRIGGER:TIMER 5ms"
PRINT@DSO;":TRIG:TIM 5E-3"
```

:TRIGger:TIMer?

本クエリで、現在のイベントバリュー設定を読み出します。

クエリシンタックス　:TRIGger:TIMer?
ニーモニック　TIMER?
省略形　TIM?

応答メッセージ　:TRIGger:TIMer <event_timer>

・「:TRIGger:TIMer」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<event_timer> は＜NR3 数値応答データ＞フォーマットの数値です。
・単位を s とみなして数値を計算しますが、サフィックス部の文字は送信しません。

3.6.13 コンビネーショントリガ専用コマンド

トリガモード(:TRIGger:MODE)設定=COMBination の場合にのみ使用するコマンド群を規定します。

エラー処理　　トリガモード(:TRIGger:MODE)設定≠COMBination の場合には、CCHannel、WHEN、LSTateの 3 種のコマンド/クエリは無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

3.6.13.1 CCHannel コマンド/ CCHannel? クエリ

:TRIGger:CCHannel <clock_ch_arg>

本コマンドでコンビネーショントリガ用のクロックチャネルを設定します。

エラー処理　　CCHannel コマンド/クエリ共、トリガモード(:TRIGger:MODE)設定≠COMBination の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TRIGger:CCHannel <clock_ch_arg>
ニーモニック　CCHANNEL
省略形　CCH
・<clock_ch_arg> はクロックチャネル設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・CHannel1、CHannel2、CHannel3、CHannel4、OFF のいずれかとします。

例　　PRINT@DSO;":TRIGGER:CCHANNEL CHANNEL1"
PRINT@DSO;":TRIG:CCH CH2"

:TRIGger:CCHannel?

本クエリで、現在のクロックチャネル設定を読み出します。

クエリシンタックス　:TRIGger:CCHannel?
ニーモニック　CCHANNEL?
省略形　CCH?

応答メッセージ　:TRIGger:CCHannel <clock_ch>
・「:TRIGger:CCHannel」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<clock_ch> は＜文字応答データ＞フォーマットのニーモニックです。
・CHANnel1、CHANnel2、CHANnel3、CHANnel4、OFF のいずれかです。

3.6.13.2 WHEN コマンド / WHEN? クエリ

:TRIGger:WHEN <when_arg>

本コマンドでトリガモード(:TRIGger:MODE)設定=COMBination の場合の WHEN を設定します。

エラー処理　　WHEN コマンド/クエリ共、トリガモード(:TRIGger:MODE)設定≠COMBination の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TRIGger:WHEN <when_arg>
ニーモニック　WHEN
省略形　WHEN(省略形なし)
・<when_arg> は WHEN 設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・EQUal または NEQual のいずれかとします。
・ローカルメニュー表示の ＝ が EQUal に、≠ が NEQual に相当します。

例　　PRINT@DSO;":TRIGGER:WHEN EQUAL"
PRINT@DSO;":TRIG:WHEN EQU"

:TRIGger:WHEN?

本クエリで、現在の WHEN 設定を読み出します。

クエリシンタックス　:TRIGger:WHEN?
ニーモニック　WHEN?
省略形　WHEN?(省略形なし)

応答メッセージ　:TRIGger:WHEN <when>
・「:TRIGger:WHEN」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<when> は＜文字応答データ＞フォーマットのニーモニックです。
・EQUal または NEQual のいずれかです。

3.6.13.3 LSTate コマンド/ LSTate? クエリ

:TRIGger:LSTate <logic_state_arg>

本コマンドでトリガモード(:TRIGger:MODE)設定=COMBination の場合に、:TRIGger:PATH 指定されたチャネルのステートを設定します。

エラー処理
- LSTATE コマンド/クエリ共、トリガモード(:TRIGger:MODE)設定≠COMBinationの場合には実行エラーとします。
- :TRIGger:PATH指定=LINE の場合も実行エラーとします。
- :TRIGger:PATH 指定されたチャネルがクロックチャネルに指定されている場合も実行エラー(SLOPE コマンドを使用すべき)とします。
- いずれの場合も、以下のメッセージをエラーキューに格納します。
  -221,"Setting conflict"

コマンドシンタックス :TRIGger:LSTate <logic_state_arg>
ニーモニック LSTATE
省略形 LST
- <logic_state_arg> はステート設定パラメータで、<文字プログラムデータ>フォーマットのニーモニックです。
- HIGH、LOW、DONTcare のいずれかとします。
  但し、H、L、X も有効とします。

例
```
PRINT@DSO;":TRIGGER:LSTATE HIGH"
PRINT@DSO;":TRIG:LST H"
```

:TRIGger:LSTate?

本クエリで、現在のステート設定を読み出します。

クエリシンタックス :TRIGger:LSTate?
ニーモニック LSTATE?
省略形 LST?

応答メッセージ :TRIGger:LSTate <logic_state>
- 「:TRIGger:LSTate」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
- <logic_state> は<文字応答データ>フォーマットのニーモニックです。
- HIGH、LOW、DONTcare のいずれかです。

3.6.14 TV/HDTV TRIGGER 専用コマンド

トリガモード(:TRIGger:MODE)設定=TV または HDTV の場合にのみ使用するコマンド群を規定します。

エラー処理　　DS-9121 で TV/HDTV トリガ(オプション)が実装されていない場合、TVSelect、TVLine、TVMode の 3 種のコマンド/クエリは無効で、コマンドエラーとし、以下のメッセージをエラーキューに格納します。
-141,"Invalid character data"

3.6.14.1 TVSelect コマンド/ TVSelect? クエリ

:TRIGger:TVSelect <tv_select_arg>

本コマンドでトリガモード(:TRIGger:MODE)設定=TV または HDTV の TV セレクトを設定します。

エラー処理　　DS-9121 で TVSELECT コマンド/クエリ共、トリガモード(:TRIGger:MODE)設定が TV でも HDTV でもない場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:TRIGger:TVSelect <tv_select_arg>
ニーモニック　TVSELECT
省略形　TVS
・<tv_select_arg> は TVセレクト設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・ODD、EVEN、BOTH、TVH のいずれかとします。

例　　PRINT@DSO;":TRIGGER:TVSELECT BOTH"
PRINT@DSO;":TRIG:TVS ODD"

:TRIGger:TVSelect?

本クエリで現在の TV セレクトの設定を読み出します。

クエリシンタックス　:TRIGger:TVSelect?
ニーモニック　TVSELECT?
省略形　TVS?

応答メッセージ　:TRIGger:TVSelect <tv_select>
・「:TRIGger:TVSelect」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<tv_select> は＜文字応答データ＞フォーマットのニーモニックです。
・ODD、EVEN、BOTH、TVH のいずれかとします。

3.6.14.2 TVLine コマンド/ TVLine? クエリ

:TRIGger:TVLine <tv_line_arg>

本コマンドでトリガモード(:TRIGger:MODE)設定=TV または HDTV の TV ライン数を設定します。

エラー処理
- トリガモード(:TRIGger:MODE)設定が TV でも HDTV でもない場合は TVLine コマンド/クエリは無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
  -221,"Setting conflict"
- <tv_line_arg> として負の数を設定しようとした場合は実行エラーとし、以下のメッセージをエラーキューに格納します。
  -222,"Data out of range"

コマンドシンタックス :TRIGger:TVLine <tv_line_arg>
ニーモニック TVLINE
省略形 TVL
<tv_line_arg> は TV ライン数設定用のパラメータで＜数値プログラムデータ＞フォーマットの数値とします。

例
```
PRINT@DSO;":TRIGGER:TVLINE 40"
PRINT@DSO;":TRIG:TVL 4e+1"
```

:TRIGger:TVLine?

本クエリで、現在の TV ライン数の設定を読み出します。

クエリシンタックス :TRIGger:TVLine?
ニーモニック TVLINE?
省略形 TVL?

応答メッセージ :TRIGger:TVLine <tv_line>
- 「:TRIGger:TVLine」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
- <tv_line> は＜NR1 数値応答データ＞フォーマットの整数です。

3.6.14.3 TVMode コマンド/ TVMode? クエリ

:TRIGger:TVMode <tv_mode_arg>

本コマンドで TV モードを設定します。

コマンドシンタックス　:TRIGger:TVMode <tv_mode_arg>
ニーモニック　TVMODE
省略形　TVM
・<tv_mode_arg> は TV モード設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・NTSC、PAL のいずれかとします。

例　PRINT@DSO;":TRIGGER:TVMODE NTSC"
PRINT@DSO;":TRIG:TVM PAL"

:TRIGger:TVMode?

本クエリで、現在の TV モードの設定を読み出します。

クエリシンタックス　:TRIGger:TVMode?
ニーモニック　TVMODE?
省略形　TVM?

応答メッセージ　:TRIGger:TVMode <tv_mode>
・「:TRIGger:TVMode」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<tv_mode> は＜文字応答データ＞フォーマットのニーモニックです。
・NTSC、PAL のいずれかとします。

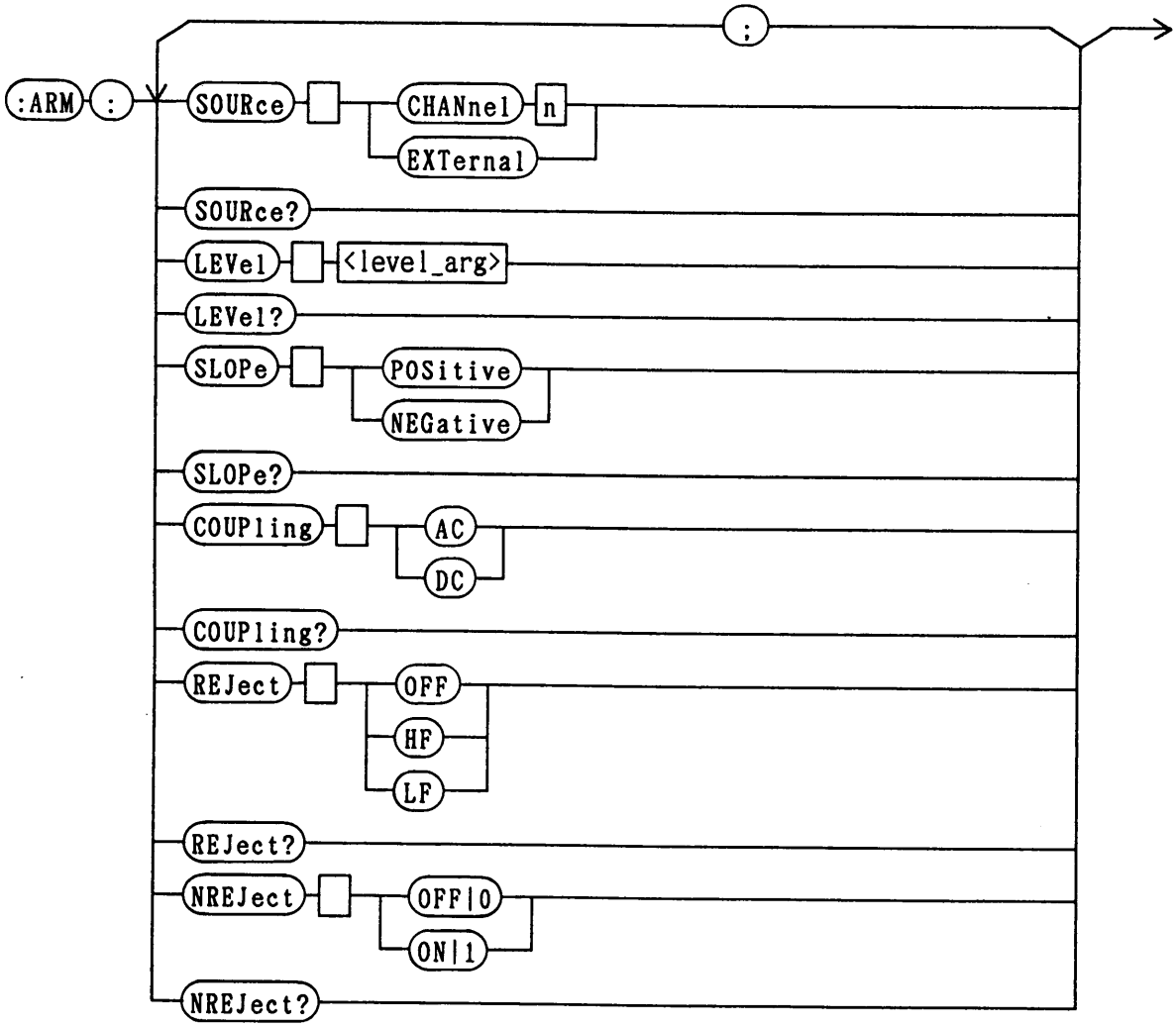
3.7 ARM 関連コマンド

:ARM:<item>…… と指定すると、イベントトリガのアーム信号に関する各種設定、確認ができます。実際の設定項目は <item> 部のヘッダにより指定されます。

エラー処理　　ARM 親コマンド下の全サブコマンド/クエリとも、トリガモードがイベントの場合のみ有効で、それ以外の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:ARM:<item>……
ニーモニック　ARM
省略形　　ARM(省略形なし)

図3.7.1 ARM 関連コマンド全体像



3.7.1 SOURce コマンド/ SOURce? クエリ

:ARM:SOURce <arm_source_arg>

本コマンドでイベントトリガ用のアームソースチャネル(ARM SOURCE CHANNEL)の設定をします。

コマンドシンタックス　:ARM:SOURce <arm_source_arg>
　　ニーモニック　SOURCE
　　省略形　SOUR
・<arm_source_arg> はアームソースチャネル設定パラメータで、<文字プログラムデータ>フォーマットのニーモニックです。
・CHannel1、CHannel2、CHannel3、CHannel4、EXTernal のいずれかとします。

例　PRINT@DSO;":ARM:SOURCE CHANNEL1"
　　PRINT@DSO;":ARM:SOUR CH2"

:ARM:SOURce?

本クエリで、現在のアームソースチャネル設定を読み出します。

クエリシンタックス　:ARM:SOURce?
　　ニーモニック　SOURCE?
　　省略形　SOUR?

応答メッセージ　:ARM:SOURce <arm_source>
・「:ARM:SOURce」部分は :SYSTEM:HEADER 設定=OFF の場合には省略します。
・<arm_source> は現在のアームソースチャネル設定を示す<文字応答データ>フォーマットのニーモニックです。
・CHANnel1、CHANnel2、CHANnel3、CHANnel4、EXTernal のいずれかとなります。

3.7.2 LEVel コマンド/ LEVel? クエリ

:ARM:LEVel <arm_level_arg>

本コマンドでアームソースに指定されたチャネルのトリガレベルを設定します。

エラー処理　　アームソース(:ARM:SOURce)設定＝EXTernalの場合はトリガレベル設定は無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:ARM:LEVel <arm_level_arg>
ニーモニック　LEVEL
省略形　LEV

・<arm_level_arg> はトリガレベル設定パラメータで、＜数値プログラムデータ＞フォーマットを基本とし、サフィックス部も有効とします。
・デフォルトのサフィックス乗数は×10^0、デフォルトのサフィックス単位は V とします。
・設定可能範囲は、そのときの垂直軸感度(VOLTS/DIV)設定の ±16div に相当する電圧値です。

エラー処理　　<arm_level_arg> のサフィックス単位としてはでは V のみ有効で、これ以外の文字を受け取った場合はコマンドエラーとし、以下のメッセージをエラーキューに格納します。
-131,"Invalid suffix"

例
```
PRINT@DSO;":ARM:LEVEL 500mV"
PRINT@DSO;":ARM:LEV 5E-1"
```

:ARM:LEVel?

本クエリでアームソースに指定されたチャネルの現在のトリガレベル設定を読み出します。

クエリシンタックス　:ARM:LEVel?
　　ニーモニック　LEVEL?
　　省略形　LEV?

応答メッセージ　:ARM:LEVel <arm_level>
・「:ARM:LEVel」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
・<arm_level> は ＜NR3 数値応答データ＞フォーマットで、トリガレベルを示します。
・単位を V とみなした数値を送信しますが、サフィックス部は送信しません。
・:ARM:SOURce設定=EXTernal の場合には以下の応答データとします。
1.5E+0

3.7.3 SLOPe コマンド/ SLOPe? クエリ

:ARM:SLOPe <arm_slope_arg>

本コマンドでアームソースに設定されたチャネルのトリガスロープを設定します。

コマンドシンタックス　:ARM:SLOPe <arm_slope_arg>
　　ニーモニック　SLOPE
　　省略形　SLOP
　　・<arm_slope_arg> はトリガスロープ設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
　　・POSitive または NEGative とします。
　　・POSitive がトリガメニュー表示の + に、NEGative が - に相当します。

例　PRINT@DSO;":ARM:SLOPE POSITIVE"
　　PRINT@DSO;":ARM:SLOP POS"

:ARM:SLOPe?

本クエリでアームソースに設定されたチャネルの現在のトリガスロープの設定を読み出します。

クエリシンタックス　:ARM:SLOPe?
　　ニーモニック　SLOPE?
　　省略形　SLOP?

応答メッセージ　:ARM:SLOPe <arm_slope>
　　・「:ARM:SLOPe」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
　　・<arm_slope> は＜文字応答データ＞フォーマットのニーモニックです。
　　・POSitive または NEGative のいずれかです。

3.7.4 COUPling コマンド/ COUPling? クエリ

:ARM:COUPling <arm_coupling_arg>

本コマンドでアームソースに設定されたチャネルの同期結合方式の設定をします。

エラー処理　　アームソース(:ARM:SOURce)設定=EXTernal の場合は COUPling サブコマンドによる設定は無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:ARM:COUPling <arm_coupling_arg>
ニーモニック　COUPLING
省略形　COUP
・<arm_coupling_arg> は同期結合方式設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・AC または DC のいずれかとします。

例　PRINT@DSO;":ARM:COUPLING DC"
PRINT@DSO;":ARM:COUP dc"

:ARM:COUPling?

本クエリでアームソースに設定されたチャネルの現在の同期結合方式の設定を読み出します。

クエリシンタックス　:ARM:COUPling?
ニーモニック　COUPLING?
省略形　COUP?

応答メッセージ　:ARM:COUPling <arm_coupling>
・「:ARM:COUPling」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
・<arm_coupling> は＜文字応答データ＞フォーマットのニーモニックです。
・AC または DC のいずれかです。

3.7.5 REJect コマンド/ REJect? クエリ

:ARM:REJect <arm_rej_arg>

本コマンドでアームソースに設定されたチャネルの HF/LF リジェクションを設定します。

エラー処理　　アームソース(:ARM:SOURce)設定=EXTernal の場合は REJect サブコマンドによる設定は無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:ARM:REJect <arm_rej_arg>
ニーモニック　REJECT
省略形　REJ
・<arm_rej_arg> は HF/LF リジェクション設定パラメータで、<文字プログラムデータ>フォーマットのニーモニックです。
・OFF,HF,LF のいずれかとします。

例　PRINT@DSO;":ARM:REJECT HF"
PRINT@DSO;":ARM:REJ hf"

:ARM:REJect?

本クエリでアームソースに設定されたチャネルの HF/LF リジェクションの現在の設定を読み出します。

クエリシンタックス　:ARM:REJect?
ニーモニック　REJECT?
省略形　REJ?

応答メッセージ　:ARM:REJect <arm_rej>
・「:ARM:REJect」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
・<arm_rej> は<文字応答データ>フォーマットのニーモニックです。
・OFF,HF,LF のいずれかです。

3.7.6 NREJect コマンド/ NREJect? クエリ

:ARM:NREJect <mode_arg>

本コマンドでアームソースに設定されたチャネルのノイズリジェクションを設定します。

エラー処理　　アームソース(:ARM:SOURce)設定=EXTernal の場合は NREJect サブコマンドによる設定は無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:ARM:NREJect <mode_arg>
ニーモニック　NREJECT
省略形　NREJ

・<mode_arg> はノイズリジェクション設定パラメータで、＜文字プログラムデータ＞フォーマットの ON または OFF、＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
・数値を丸めて整数化した後の値が 0 の場合はOFF、0 以外の場合は ON とみなします。

例
```
PRINT@DSO;":ARM:NREJECT ON"
PRINT@DSO;":ARM:NREJ 1"
```

:ARM:NREJect?

本クエリでアームソースに設定されたチャネルのノイズリジェクションの現在の設定を読み出します。

クエリシンタックス　:ARM:NREJect?
ニーモニック　NREJECT?
省略形　NREJ?

応答メッセージ　:ARM:NREJect <mode>

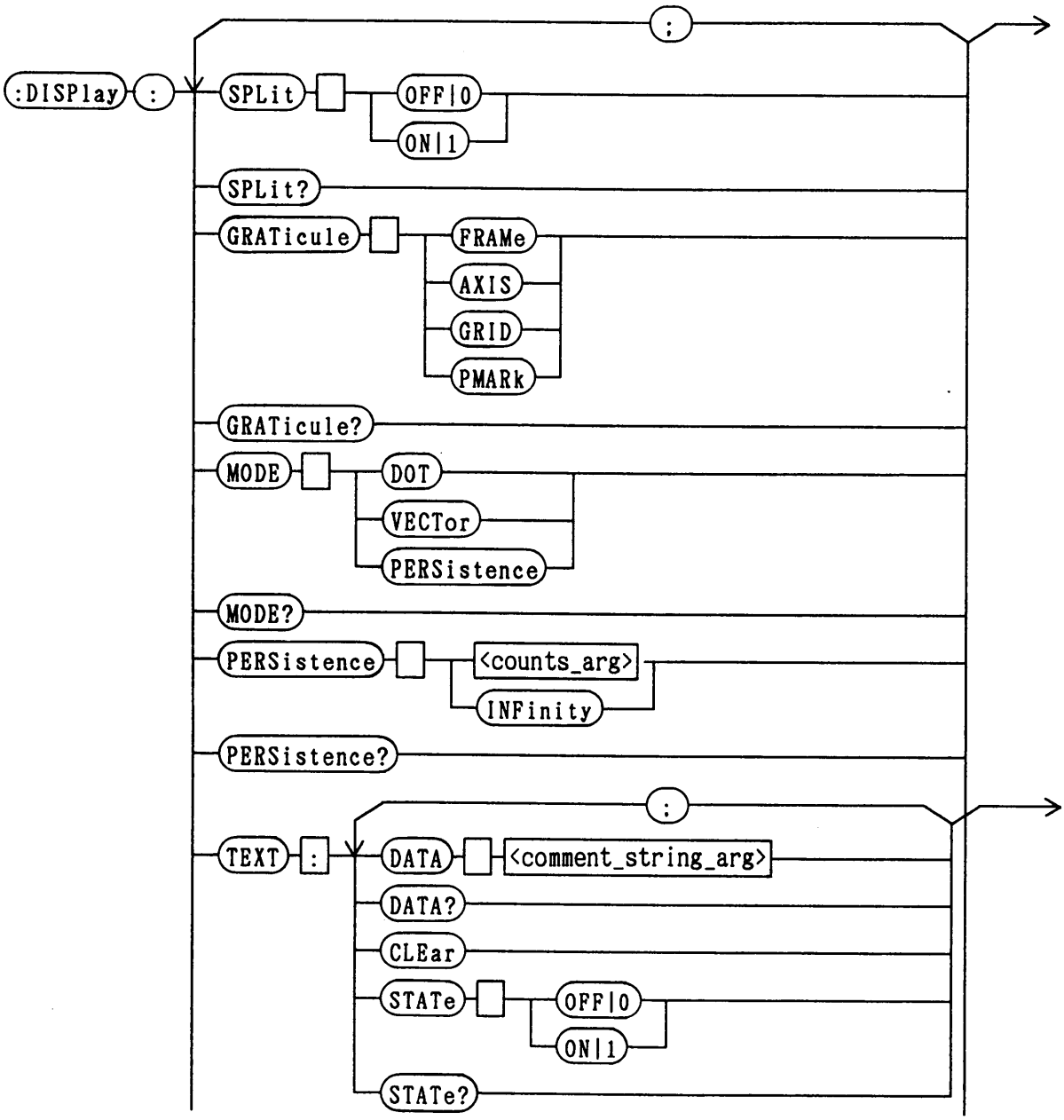
・「:ARM:NREJect」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットの数値で、0 または 1 の 1 桁の数字とします。
・0 が OFF、1 が ON を意味します。

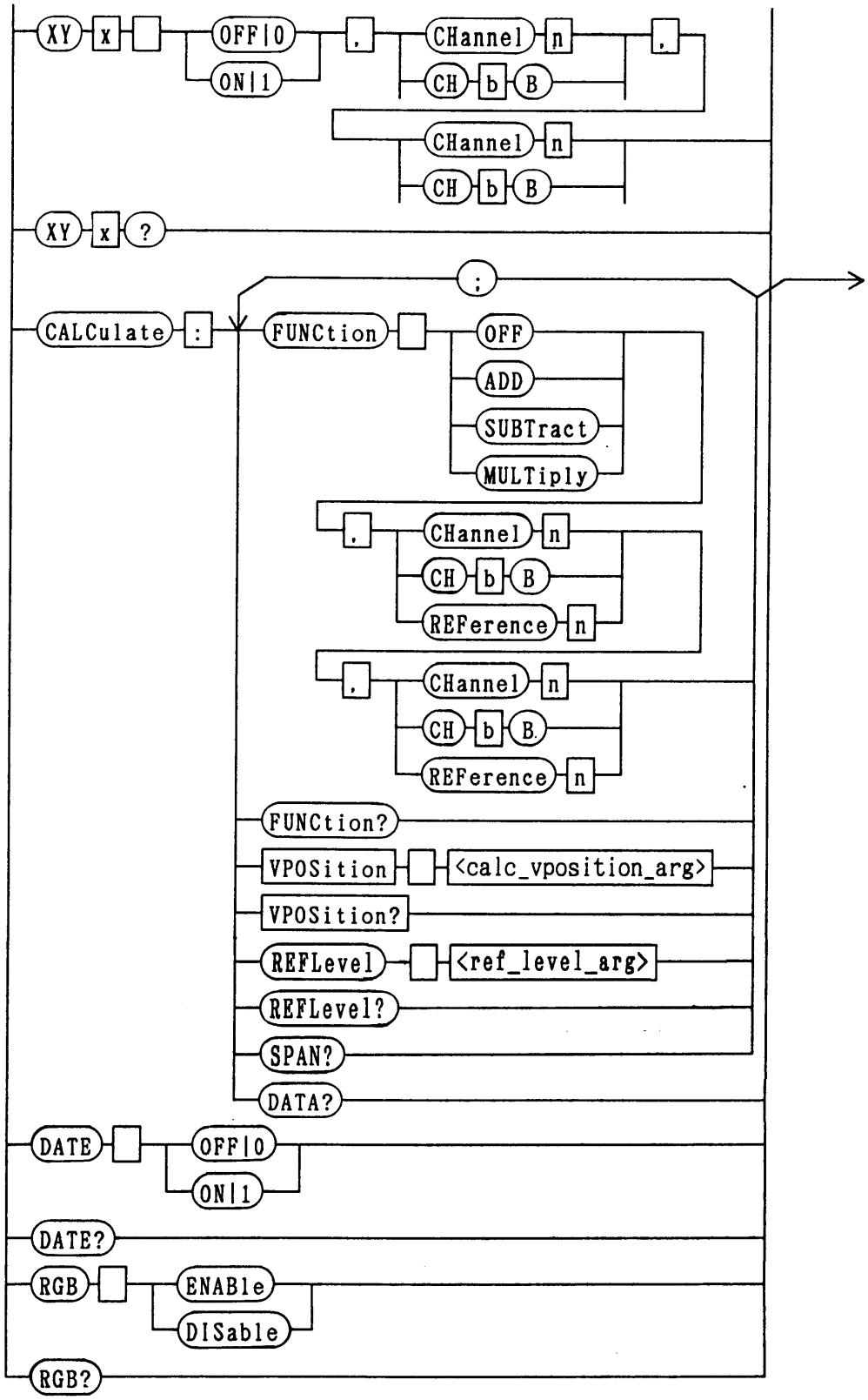
3.8 DISPlay 関連コマンド

:DISPlay:<item>……　と指定すると、ディスプレー関連の各種設定が可能です。
実際の設定項目は <item> 部のヘッダにより指定されます。
なお、リファレンス波形に関する項目は REF 親コマンド下に独立させます。

コマンドシンタックス　:DISPlay:<item>……
　　ニーモニック　DISPLAY
　　省略形　　DISP

図3.8.1 DISPlay 関連コマンド全体像




XY
x
OFF|0
ON|1
,
CHannel
n
CH
b
B
XY
x
?
;
CALCulate
:
FUNCtion
OFF
ADD
SUBTract
MULTiply
REFerence
FUNCtion?
VPOSition
<calc_vposition_arg>
VPOSition?
REFLevel
<ref_level_arg>
REFLevel?
SPAN?
DATA?
DATE
DATE?
RGB
ENABle
DISable
RGB?

3.8.1 SPLit コマンド/ SPLit? クエリ

:DISPlay:SPLit <mode_arg>

本コマンドでスプリット表示(ON/OFF)を設定します。

エラー処理　　:DISPlay:SPLit コマンド/クエリとも、メモリ使用種別(:ACQuire:MEMory)設定=ROLL の場合には無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:DISPlay:SPLit <mode_arg>
ニーモニック　SPLIT
省略形　SPL

・<mode_arg> はスプリット表示(ON/OFF)設定用のパラメータで、＜文字プログラムデータ＞フォーマットの ON または OFF、＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

例
```
PRINT@DSO;":DISPLAY:SPLIT 1"
PRINT@DSO;":DISP:SPL ON"
```

:DISPlay:SPLit?

本クエリで、現在のスプリット表示の設定を読み出します。

クエリシンタックス　:DISPlay:SPLit?
ニーモニック　SPLIT?
省略形　SPL?

応答メッセージ　:DISPlay:SPLit <mode>

・「:DISPlay:SPLit」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の1桁の数字とします。
・0 が OFF、1 が ON を意味します。

3.8.2 GRATicule コマンド/ GRATicule? クエリ

:DISPlay:GRATicule <graticule_arg>

本コマンドでグラテキュール種別を設定します。

コマンドシンタックス　:DISPlay:GRATicule <graticule_arg>
ニーモニック　GRATICULE
省略形　GRAT

・<graticule_arg> はグラテキュール種別設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・FRAMe、AXIS、GRID、PMARk のいずれかとします。

例
```
PRINT@DSO;":DISPLAY1:GRATICULE FRAME"
PRINT@DSO;":DISP1:GRAT fram"
```

:DISPlay:GRATicule?

本クエリで現在のグラテキュール種別の設定状態を読み出します。

クエリシンタックス　:DISPlay:GRATicule?
ニーモニック　GRATICULE?
省略形　GRAT?

応答メッセージ　:DISPlay:GRATicule <graticule>

・「:DISPlay:GRATicule」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<graticule> は＜文字応答データ＞フォーマットのニーモニックです。
・FRAMe、AXIS、GRID、PMARk のいずれかとなります。

3.8.3 MODE コマンド/ MODE? クエリ

:DISPlay:MODE <display_mode_arg>

本コマンドで ディスプレイモードを設定します。

コマンドシンタックス　:DISPlay:MODE <display_mode_arg>
　　ニーモニック　MODE
　　省略形　MODE(省略形なし)
　　・<display_mode_arg>ディスプレイモード設定用のパラメータで、<文字プログラムデータ>フォーマットのニーモニックです。
　　・DOT、VECTor、PERSistence のいずれかとします。

エラー処理　・PERSistence の指定はメモリ使用種別(:ACQuire:MEMory設定)がショート(SHORt)でのみ有効で、それ以外では実行エラーとなりエラーキューに以下のメッセージを格納します。
　　-221,"Setting conflict"

:DISPlay:MODE?

本クエリで 現在のディスプレイモードの設定状態を読み出します。

クエリシンタックス　:DISPlay:MODE?
　　ニーモニック　MODE?
　　省略形　MODE?(省略形なし)

応答メッセージ　:DISPlay:MODE <display_mode>
　　・「:DISPlay:MODE」部分は:SYSTem:HEADer設定=OFF の場合には省略します。
　　・<display_mode>は<文字応答データ>フォーマットのニーモニックです。
　　・DOT、VECTor、PERSistence のいずれかになります。

3.8.4 PERSistence コマンド/ PERSistence? クエリ

:DISPlay:PERSistence <count_arg>

本コマンドでパーシスタンス回数を設定します。

コマンドシンタックス　:DISPlay:PERSistence <count_arg>
　　ニーモニック　PERSISTENCE
　　省略形　PERS
・<count_arg> はパーシスタンス回数設定用のパラメータで、基本的には<数値プログラムデータ>フォーマットの数値とします。
・設定可能な数値は 2 のベキ乗ですので、数値を丸めて整数化した後の値に一番近い内部設定可能値に飛びつくものとします。
なお、<文字プログラムデータ>の INFinity を例外的に許し、数値 9.9E+37 と共に、無限大設定とみなします。

エラー処理　:DISPlay:PERSistence コマンド/クエリとも、メモリ使用種別(:ACQuire :MEMory)=SHORt モードでのみ有効で、それ以外では実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例
```
PRINT@DSO;":DISPLAY:PERSISTENCE 1"
PRINT@DSO;":DISP:PERS INF"
```

:DISPlay:PERSistence?

本クエリで、現在のパーシスタンス回数設定値を読み出します。

クエリシンタックス　:DISPlay:PERSistence?
　　ニーモニック　PERSISTENCE?
　　省略形　PERS?

応答メッセージ　:DISPlay:PERSistence <count>
・「:DISPlay:PERSistence」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<count> は<NR3 数値応答データ>フォーマットとし、パーシスタンス回数設定値を表す数字とします。
・無限大の場合には 9.9E+37 を応答データとします。

3.8.5 TEXT コマンド

:DISPlay:TEXT:<item>……　と指定すると、コメント文字列を対象とする各種設定が可能です。
実際の設定項目は、TEXT サブコマンドの後に続く <item> 部のヘッダにより指定されます。

コマンドシンタックス　:DISPlay:TEXT:<item>……
　　ニーモニック　TEXT
　　省略形　　TEXT(省略形なし)

3.8.5.1 DATA コマンド/ DATA? クエリ

:DISPlay:TEXT:DATA <string_arg>

本コマンドでコメント文字列を設定します。

コマンドシンタックス　:DISPlay:TEXT:DATA <string_arg>
　　ニーモニック　DATA
　　省略形　　DATA(省略形なし)
　　・<string_arg> はコメント文字列設定用のパラメータで、＜文字列プログラムデータ＞フォーマットの ASCII 文字列とします。
　　・文字列プログラムデータフォーマットとは、ダブルクォート(")あるいはシングルクォート(')文字によってはさまれた ASCII 文字列のことです。
　　・<string_arg> 内の ASCII 文字の MSB(ビット7)をキャンセル(0)し、さらに本器で表示不可能な文字(コード 0～1FH)はスペース(20H)に置き換えた後に評価します。この場合には、小文字は小文字として認識します。

エラー処理　　後側のクォート文字の前に END メッセージを受信した場合は、コマンドエラーとし、以下のメッセージをエラーキューに格納します。
　　-151,"Invalid String data"

例　　PRINT@DSO;":DISPLAY:TEXT:DATA 'abcdefghijklmnopqrstuvzxyz'"
　　PRINT@DSO;":DISP:TEXT:DATA 'xyz'"

:DISPlay:TEXT:DATA?

本クエリで、現在のコメント文字列を読み出します。

クエリシンタックス　:DISPlay:TEXT:DATA?
　　ニーモニック　DATA?
　　省略形　　DATA?(省略形なし)

応答メッセージ　　:DISPlay:TEXT:DATA <string>
　　・「:DISPlay:TEXT:DATA」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<string> は＜文字列応答データ＞フォーマットとし、現在のコメント文字列をダブルクォート(")で囲んだものとなります。

3.8.5.2 CLEAR コマンド

:DISPlay:TEXT:CLEAR コマンドでコメント文字列の行クリアを実施します。

コマンドシンタックス :DISPlay:TEXT:CLEar
ニーモニック CLEAR
省略形 CLE

例
```
PRINT@DSO;":DISPLAY:TEXT:CLEAR"
PRINT@DSO;":DISP:TEXT:CLE"
```

3.8.5.3 TEXT:STATe コマンド/ TEXT:STATe? クエリ

:DISPlay:TEXT:STATe <mode_arg>

本コマンドでコメント表示(ON/OFF)を設定します。

コマンドシンタックス :DISPlay:TEXT:STATe <mode_arg>
ニーモニック STATE
省略形 STAT

・<mode_arg> はコメント表示(ON/OFF)設定用のパラメータで、＜文字プログラムデータ＞フォーマットの ON または OFF、＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

例
```
PRINT@DSO;":DISPLAY:STATE 1"
PRINT@DSO;":DISP:STAT ON"
```

:DISPlay:TEXT:STATe?

本クエリで、現在のコメント表示の設定を読み出します。

クエリシンタックス :DISPlay:TEXT:STATe?
ニーモニック STATE?
省略形 STAT?

応答メッセージ :DISPlay:TEXT:STATe <mode>

・「:DISPlay:TEXT:STATe」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の1桁の数字とします。
・0 が OFF、1 が ON を意味します。

3.8.6 XY<x> コマンド/ XY<x>? クエリ

:DISPlay:XY<x> <mode_arg>,<x_source_srg>,<y_source_srg>

本コマンドで X-Y 波形(そのx)の ON/OFF と X軸/Y軸ソースチャンネルの設定をします。

コマンドシンタックス :DISPlay:XY<x> <mode_arg>,<x_source_srg>,<y_source_srg>

ニーモニック XY1
XY2

省略形 XY1(省略形なし)
XY2(省略形なし)

エラー処理

- <x>は数値ではなく、ニーモニックを構成する ASCII 数字文字1文字です。従って、<x>が 1 でも 2 でもない場合や、XYと<x>の間に空白文字などがある場合、<x>が 01 などと書かれている場合には、コマンドエラーとし、以下のメッセージをエラーキューに格納します。
  -113,"Undefined header"
- :DISPlay:XY<x> コマンド/クエリとも、メモリ使用種別(:ACQuire:MEMory)設定=SHORt の場合のみ有効で、それ以外では実行エラーとし、以下のメッセージをエラーキューに格納します。
  -221,"Setting conflict"
- <mode_arg> はXY<x>表示(ON/OFF)設定用のパラメータで、<文字プログラムデータ>フォーマットの ON または OFF、<数値プログラムデータ>フォーマットの数値のいずれかとします。
- 数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。
- <x_source_srg> はX<x>ソースチャネル、<y_source_srg> はY<x>ソースチャネル設定パラメータで、いずれも<文字プログラムデータ>フォーマットのニーモニックです。
- CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B のいずれかとします。
- CH1B、CH2B 指定は、B掃引(:TIMebase:BSWeep)設定=OFF の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
  -221,"Setting conflict"

例

```
PRINT@DSO;":DISPLAY:XY1 ON,CHANNEL1,CHANNEL2"
PRINT@DSO;":DISP:XY2 1,CH1,CH2"
```

:DISPlay:XY<x>?

本クエリで現在の X-Y 波形(そのx)の設定を読み出します。

クエリシンタックス　:DISPlay:XY<x>?

ニーモニック　XY1?
XY2?

省略形　XY1?(省略形なし)
XY2?(省略形なし)

応答メッセージ　:DISPlay:XY<x> <mode>,<x_source>,<y_source>

・「:DISPlay:XY<x>」部分は:SYSTem:HEADer 設定=OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の 1桁の数字とします。
・0 が OFF、1 が ON を意味します。
・<x_source>,<y_source> は X軸/Y軸ソースチャンネルを表すパラメータで、いずれも＜文字プログラムデータ＞フォーマットのニーモニックです。
・CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B のいずれかとします。

3.8.7 CALCulate コマンド/ CALCulate? クエリ

:DISPlay:CALCulate:<item>…… と指定すると、波形演算機能を対象とする各種設定が可能です。
実際の設定項目は CALCulate サブコマンドの後に続く <item> 部のヘッダにより指定されます。

コマンドシンタックス :DISPlay:CALCulate:<item>……
ニーモニック CALCULATE
省略形 CALC

3.8.7.1 FUNCtion コマンド/ FUNCtion? クエリ

:DISPlay:CALCulate:FUNCtion <operator_arg>,<op1_arg>,<op2_arg>

本コマンドで波形演算の演算子と演算対象波形の設定をします。

コマンドシンタックス :DISPlay:CALCulate:FUNCtion <operator_arg>,<op1_arg>,<op2_arg>
ニーモニック FUNCTION
省略形 FUNC

エラー処理
- :DISPlay:CALCulate:FUNCtion コマンド/クエリとも、メモリ使用種別 (:ACQuire:MEMory) 設定=SHORt 以外では無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
  -221,"Setting conflict"
- <operator_arg> は演算子設定用のパラメータで、<文字プログラムデータ>フォーマットのニーモニックです。
- OFF、ADD、SUBTract、MULTiply、SPECtrum のいずれかとします。
- <op1_arg> は演算対象波形 (その1) 、<op2_arg> は演算対象波形 (その2) 設定パラメータで、いずれも<文字プログラムデータ>フォーマットのニーモニックです。
  ただし、<operator_arg> に SPECtrum を設定したときは <op2_arg> の設定は演算には影響しません。
- CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B、REFerence1、REFerence2、REFerence3、REFerence4 のいずれかとします。

エラー処理 CH1B、CH2B 指定は B 掃引 (:TIMEBASE:BSWeep) 設定=OFF の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例
```
PRINT@DSO;":DISP:CALC:FUNCTION ADD,CHANNEL1,CHANNEL2"
PRINT@DSO;":DISP:CALC:FUNC add,CH1,CH2"
```

:DISPlay:CALCulate:FUNCtion?

本クエリで現在の波形演算の演算子と演算対象波形設定を読み出します。

クエリシンタックス :DISPlay:CALCulate:FUNCtion?
ニーモニック FUNCTION?
省略形 FUNC?

応答メッセージ :DISPlay:CALCulate:FUNCtion <operator>,<op1>,<op2>
- 「:DISPlay:CALCulate:FUNCtion」部分は :SYSTem:HEADer 設定=OFF の場合には省略します。
- <operator> は演算子を表すパラメータで、<文字応答データ>フォーマットのニーモニックです。

・OFF、ADD、SUBTract、MULTiply、SPECtrum のいずれかとします。
・<op1>,<op2> は演算対象波形(その1,その2)を表すパラメータで、いずれも＜文字応答データ＞フォーマットのニーモニックです。
ただし、<operator> が SPECtrum のときは <op2> の応答メッセージは意味がありません。
・CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B、REFerence1、REFerence2、REFerence3、REFerence4 のいずれかとします。

3.8.7.2 VPOSition コマンド/ VPOSition? クエリ

:DISPlay:CALCulate:VPOSition <vpos_div_arg>

本コマンドで演算波形垂直軸ポジションを設定します。

コマンドシンタックス　:DISPlay:CALCulate:VPOSition <vpos_div_arg>
ニーモニック　VPOSITION
省略形　VPOS

エラー処理
・:DISPlay:CALCulate:VPOSition コマンド/クエリとも、メモリ使用種別(:ACQuire:MEMory)設定=SHORt モード以外では無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"
・<vpos_div_arg> は演算波形垂直軸ポジション設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値とします。

注　意
設定値は div値とみなして小数点以下 2 桁までの数値に丸めて解釈します。但し、サフィックス部の文字を許しません。

例
```
PRINT@DSO;":DISPLAY:CALCULATE:VPOSITION 4.00"
PRINT@DSO;":DISP:CALC:VPOS 4.00E-0"
```

:DISPlay:CALCulate:VPOSition?

本クエリで、現在の演算波形垂直軸ポジション設定値を読み出します。

クエリシンタックス　:DISPlay:CALCulate:VPOSition?
ニーモニック　VPOSITION?
省略形　VPOS?

応答メッセージ
:DISPlay:CALCulate:VPOSition <vpos_div>
・「:DISPlay:CALCulate:VPOSition」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<vpos_div> は＜NR3 数値応答データ＞フォーマットの数値とします。

注　意
応答データの数値は div数として計算、作成しますが、サフィックス部の文字は送信しません。

3.8.7.3 REFLevel コマンド/ REFLevel? クエリ

:DISPlay:CALCulate:REFLevel <ref_level_arg>

本コマンドでスペクトル演算波形のリファレンスレベルを設定します。

コマンドシンタックス　:DISPlay:CALCulate:REFLevel <ref_level_arg>
　　ニーモニック　REFLEVEL
　　省略形　REFL
　　・<ref_level_arg> はスペクトル演算波形のリファレンスレベル設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値とします。

エラー処理　・:DISPlay:CALCulate:REFLevel コマンド/クエリとも、メモリ使用種別(:ACQuire:MEMory) 設定=SHORt モード以外では無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
　　-221,"Setting conflict"

注　意　設定値は dBV値とみなして解釈します。
本コマンドはパラメータにサフィックス部を許しません。

例　PRINT@DSO;":DISPLAY:CALCULATE:REFLEVEL 100"
PRINT@DSO;":DISP:CALC:REFL -100"

:DISPlay:CALCulate:REFLevel?

本クエリで、現在のスペクトル演算波形のリファレンスレベル設定値を読み出します。

クエリシンタックス　:DISPlay:CALCulate:REFLevel?
　　ニーモニック　REFLEVEL?
　　省略形　REFL?

応答メッセージ　:DISPlay:CALCulate:REFLevel <ref_level>
　　・「:DISPlay:CALCulate:REFLevel」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<ref_level> は＜NR1 数値応答データ＞フォーマットの数値とします。

注　意　単位を dBV とみなした数値を送信しますが、サフィックス部は送信しません。

3.8.7.4 SPAN? クエリ

:DISPlay:CALCulate:SPAN?

本クエリで、現在のスペクトル演算波形のスパン周波数を読み出します。スパン周波数とは、画面の有効な周波数幅です。

クエリシンタックス	:DISPlay:CALCulate:SPAN?
ニーモニック	SPAN?
省略形	SPAN?（省略形なし）
応答メッセージ	:DISPlay:CALCulate:SPAN? <span_freq> ・「:DISPlay:CALCulate:SPAN」部分は :SYSTem:HEADer 設定=OFF の場合には省略します。 ・<span_freq> は＜NR3 数値応答データ＞フォーマットの数値とします。 ・正常な測定結果が得られない場合には 9.91E+37 を返します。
エラー処理	・メモリ使用種別（:ACQuire:MEMory）設定=SHORt モード以外では無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。 -221,"Setting conflict"
注　意	単位を Hz とみなした数値を送信しますが、サフィックス部は送信しません。

3.8.7.5 DATA? クエリ

:DISPlay:CALCulate:DATA?

本クエリで、演算波形データの読み出しをします。
本器が :DISPlay:CALCulate:DATA? クエリを受信した場合、ストップ状態とし、その時点で実施中の波形データのアクイジションを停止させます。本器から演算波形データの読み出しを行う場合には、基本的には、掃引方式をシングルとし、サービス要求とステータスレポート機能 または :WAIT コマンドなどを使用してアプリケーションプログラムと本器の波形データのアクイジション完了との同期をとってから、:DISPlay:CALCulate:DATA? クエリを使用してください。

注　　意

:DISPlay:CALCulate:DATA? クエリによる演算波形データの読み出し後、波形データのアクイジションを再開させるためには、:RUN コマンドや *TRG コマンド または インタフェースメッセージ GET などによる再起動を必要とします。

クエリシンタックス　:DISPlay:CALCulate:DATA?
　　ニーモニック　DATA?
　　省略形　DATA?（省略形なし）

応答メッセージ　:#800002000<word_binary_block>
・#8 は応答メッセージが<決定長任意ブロック応答データ>フォーマットであり、この後に続く 00002000 が ASCII 数字列であることを示します。
・00002000 はこの後に続く<word_binary_block> のバイト数を示します。
・<word_binary_block>は本器内部メモリの演算波形データをバイナリ値のブロックデータとしたものです。
・本器の演算波形データは 4 バイトで 1 ワードであり、下位バイトから順に転送されます。
・応答メッセージは :SYSTem:HEADer 設定に関わらずヘッダ部を含みません。

エラー処理
・メモリ使用種別（:ACQuire:MEMory）設定=SHORt モード以外では無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
　-221,"Setting conflict"
・応答メッセージがブロックデータであるため、:DISPlay:CALCulate:DATA? クエリは<プログラムメッセージ>中の最後のクエリでなくてはなりません。これに違反する場合、クエリエラーとし、:DISPlay:CALCulate:DATA? 以降のクエリに対する応答メッセージの作成、送信を行わず、以下のメッセージをエラーキューに格納します。
　-440,"Querry UNTERMINATED after indefinite response"

3.8.8 DATE コマンド/ DATE? クエリ

:DISPlay:DATE <mode_arg>

本コマンドで日付け表示(ON/OFF)を設定します。

コマンドシンタックス :DISPlay:DATE <mode_arg>
　　ニーモニック DATE
　　省略形　DATE(省略形なし)

・<mode_arg> は日付け表示 ON/OFF 設定用のパラメータで、<文字プログラムデータ>フォーマットの ON または OFF、<数値プログラムデータ>フォーマットの数値のいずれかとします。
・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

例
```
PRINT@DSO;":DISPLAY:DATE 1"
PRINT@DSO;":DISP:DATE ON"
```

:DISPlay:DATE?

本クエリで現在の日付け表示の設定を読み出します。

クエリシンタックス　:DISPlay:DATE?
　　ニーモニック　DATE?
　　省略形　DATE?(省略形なし)

応答メッセージ　:DISPlay:DATE <mode>
・「:DISPlay:DATE」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は<NR1 数値応答データ>フォーマットとし、0 または 1 の 1 桁の数字とします。
・0 が OFF、1 が ON を意味します。

3.8.9 RGB コマンド/ RGB? クエリ

:DISPlay:RGB <rgb_arg>

本コマンドで RGB 出力の設定をします。

コマンドシンタックス :DISPlay:RGB <rgb_arg>
　　ニーモニック RGB
　　省略形　RGB(省略形なし)
・<rgb_arg> は RGB 出力設定用のパラメータで、<文字プログラムデータ>フォーマットのニーモニックです。
・ENABle または DISable のいずれかとします。

例
```
PRINT@DSO;":DISPLAY1:RGB ENABLE"
PRINT@DSO;":DISP1:RGB dis"
```

:DISPlay:RGB?

本クエリで、現在の RGB 出力の設定を読み出します。

クエリシンタックス　:DISPlay:RGB?
　　ニーモニック　RGB?
　　省略形　RGB?(省略形なし)

応答メッセージ　:DISPlay:RGB <rgb>
・「:DISPlay:RGB」部分は :SYSTem:HEADer 設定=OFF の場合には省略します。
・<rgb> は<文字応答データ>フォーマットのニーモニックです。
・ENABle、DISable のいずれかとなります。

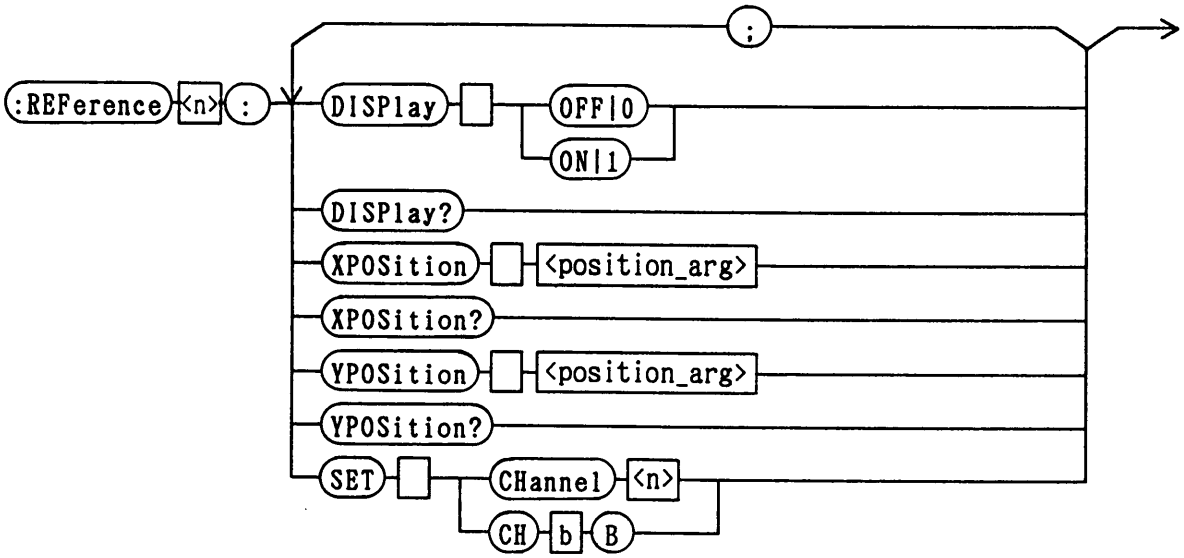
3.9 REFerence 関連コマンド

:REFerence<n>:<item>……　と指定すると、リファレンス波形を対象とする各種設定が可能です。
実際の設定項目は <item> 部のヘッダにより指定されます。

コマンドシンタックス　:REFerence<n>:<item>……
　　ニーモニック　REFERENCE1
　　REFERENCE2
　　REFERENCE3
　　REFERENCE4
　　省略形　REF1
　　REF2
　　REF3
　　REF4

エラー処理　リファレンス波形番号<n>は、数値ではなく、ニーモニックを構成するASCII 数字文字 1 文字です。従って、<n>が 5 以上の場合や、REFerenceと<n>の間にスペースなどがある場合、<n>が 01 などと書かれている場合には、コマンドエラーとし、以下のメッセージをエラーキューに格納します。
-113,"Undefined header"

図3.9.1　REFerence関連コマンド全体像



3.9.1 DISPlay コマンド / DISPlay? クエリ

:REFerence<n>:DISPlay <mode_arg>

本コマンドでリファレンス波形表示(ON/OFF)を設定します。

コマンドシンタックス　:REFerence<n>:DISPlay <mode_arg>
ニーモニック　DISPLAY
省略形　DISP

・<mode_arg> はリファレンス波形表示 ON/OFF 設定用のパラメータで、＜文字プログラムデータ＞フォーマットの ON または OFF、＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

エラー処理　:REFerence<n>:DISPlay コマンド/クエリとも、メモリ使用種別(:ACQuire:MEMory)設定=ROLL の場合には無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例
```
PRINT@DSO;":REFERENCE1:DISPLAY 1"
PRINT@DSO;":REF1:DISP ON"
```

:REFerence<n>:DISPlay?

本クエリで、現在のリファレンス波形表示設定を読み出します。

クエリシンタックス　:REFerence<n>:DISPlay?
ニーモニック　DISPLAY?
省略形　DISP?

応答メッセージ　:REFerence<n>:DISPlay <mode>

・「:REFerence<n>:DISPlay」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の 1 桁の数字とします。
・0 が OFF、1 が ON を意味します。

3.9.2 XPOSition コマンド/ XPOSition? クエリ

:REFerence<n>:XPOSition <xpos_div_arg>

本コマンドでリファレンス波形X軸ポジションを設定します。

コマンドシンタックス　:REFerence<n>:XPOSition <xpos_div_arg>
　　ニーモニック　XPOSITION
　　省略形　　XPOS
　　・<xpos_div_arg> はリファレンス波形X軸ポジション設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値とします。
　　・設定値は div数とみなして小数点以下 2 桁までの数値に丸めて解釈します。

注　意　　サフィックス部の文字を許しません。

例　　PRINT@DSO;":DISPLAY:XPOSITION 4.00"
　　PRINT@DSO;":DISP:XPOS 4.00E-0"

:REFerence<n>:XPOSition?

本クエリで、現在のリファレンス波形X軸ポジション設定値を読み出します。

クエリシンタックス　:REFerence<n>:XPOSition?
　　ニーモニック　XPOSITION?
　　省略形　　XPOS?

応答メッセージ　:REFerence<n>:XPOSition <xpos_div>
　　・「:REFerence<n>:XPOSition」部分は :SYSTem:HEADer 設定=OFF の場合には省略します。
　　・<xpos_div> は＜NR3 数値応答データ＞フォーマットの数値とします。

注　意　　応答データの数値は、div数として計算、作成しますが、サフィックス部の文字は送信しません。

3.9.3 YPOSition コマンド/ YPOSition? クエリ

:REFerence<n>:YPOSition <ypos_div_arg>

本コマンドでリファレンス波形Y軸ポジションを設定します。

コマンドシンタックス　:REFerence<n>:YPOSition <ypos_div_arg>
ニーモニック　YPOSITION
省略形　YPOS

・<ypos_div_arg> はリファレンス波形Y軸ポジション設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値とします。
・設定値は div数とみなして小数点以下 2 桁までの数値に丸めて解釈します。

注　意　サフィックス部の文字を許しません。

例
```
PRINT@DSO;":DISPLAY:YPOSITION 4.00"
PRINT@DSO;":DISP:YPOS 4.00E-0"
```

:REFerence<n>:YPOSition?

本クエリで、現在のリファレンス波形Y軸ポジション設定値を読み出します。

クエリシンタックス　:REFerence<n>:YPOSition?
ニーモニック　YPOSITION?
省略形　YPOS?

応答メッセージ　:REFerence<n>:YPOSition <ypos_div>

・「:REFerence<n>:YPOSition」部分は :SYSTem:HEADer 設定=OFF の場合には省略します。
・<ypos_div> は＜NR3 数値応答データ＞フォーマットの数値とします。

注　意　応答データの数値は、div数として計算、作成しますが、サフィックス部の文字は送信しません。

3.9.4 SET コマンド

:REFerence<n>:SET <source_arg> コマンドでリファレンス波形を作成します。

コマンドシンタックス　:REFerence<n>:SET <source_arg>

ニーモニック　SET

省略形　SET(省略形なし)

・<source_arg> はリファレンス波形作成対象チャネル設定用のパラメータで、＜文字プログラムデータ＞フォーマットののニーモニックです。

・CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B のいずれかとします。

エラー処理　CH1B、CH2B 指定は、B掃引(:TIMebase:BSWeep)設定=OFF の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。

-221,"Setting conflict"

例

```
PRINT@DSO;":REFERENCE1:SET CHANNEL1"
PRINT@DSO;":REF1:SET CH1"
```

3.9.5 DATA コマンド/ DATA? クエリ

:REFerence<n>:DATA <ref_data_arg>

本コマンドでリファレンス波形データをコントローラからの本器へ転送します。
本コマンドは基本的には :REFerence<n>:DATA? クエリで読みだしたデータを書き戻すときに使用して下さい。

コマンドシンタックス :REFerence<n>:DATA #8<block_length_byte><binary_block>
ニーモニック DATA
省略形 DATA

- #8 はプログラムデータが＜任意ブロックプログラムデータ＞フォーマットであり、この後に続く<block_length_byte>が 8 桁の ASCII 数字列であることを示します。
- <block_length_byte> は 8 桁の ASCII 数字列で、この後に続く<binary_block> のバイト数を示します。リーディングゼロをサプレスせずに、必ず 8 桁の符号なし整数とします。
- <binary_block> は、1 バイトのプリアンブルデータ(16 進数で 0A)があり、さらに続けて 500 ポイント分のバイナリデータとします。
- 1 ポイント分のデータは 2 バイト(16ﾋﾞｯﾄ)で構成され、管面センターより上側を正、下側を負とする 2 の補数形式の波形データを、下位バイト、上位バイトの順に並べたものとなります。このとき、本器の A/D 変換の分解能は 10 ビットですので、16 ビットデータ領域左詰めとし、下位 6 ビットは 0 とします。さらに、本器内部では、波形表示分解能(400ポイント)に合わせて丸めが行われます。
- X軸方向のデータの並びは表示上、左端のポイントからになります。

:REFerence<n>:DATA?

本クエリでリファレンス波形データをコントローラが本器から読み取ります。

クエリシンタックス :REFerence<n>:DATA?
ニーモニック DATA?
省略形 DATA?(省略形なし)

応答メッセージ :REFerence<n>:DATA #800001001<binary_block>

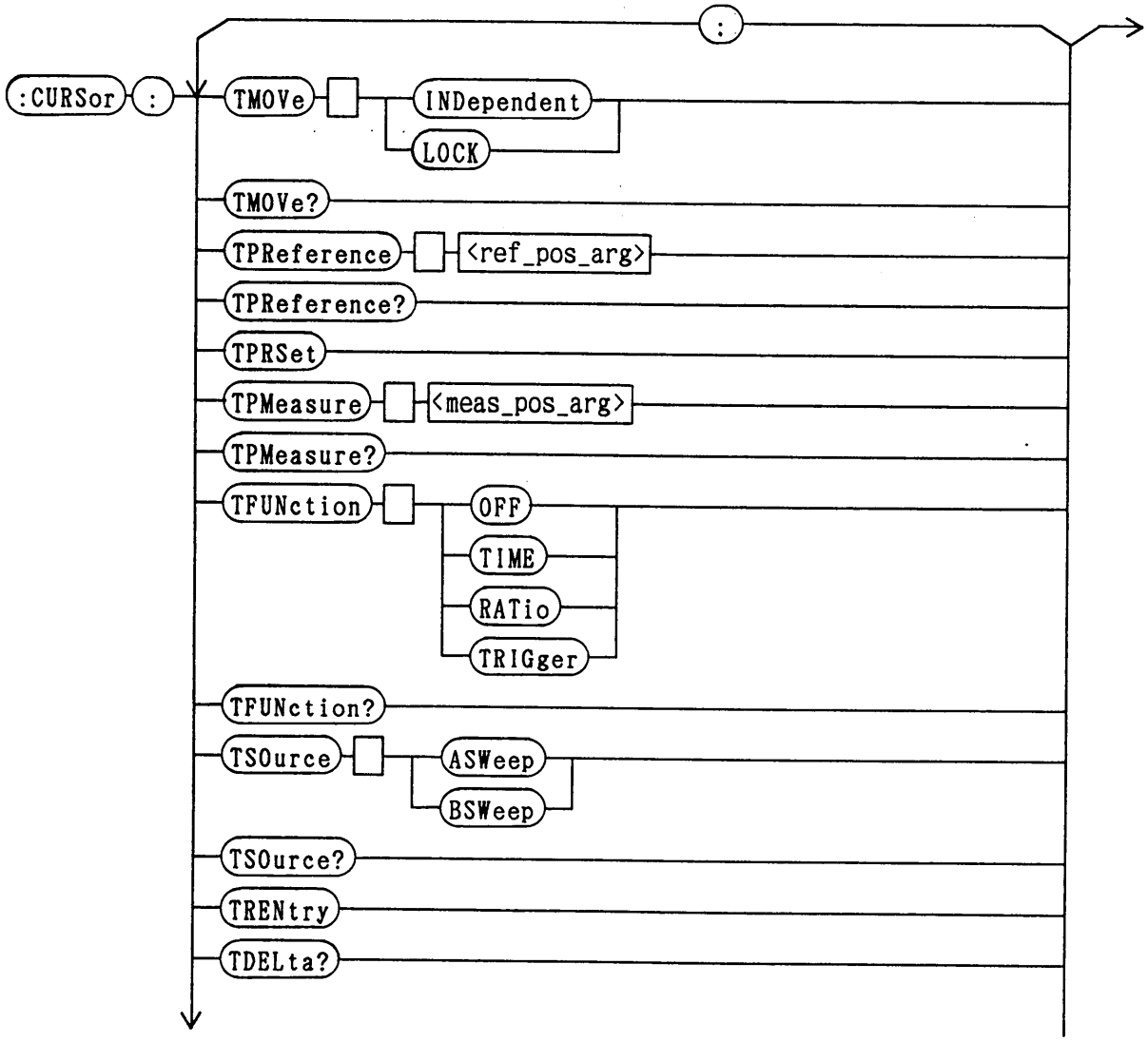
- 「:REFerence<n>:DATA」部分は :SYSTem:HEADer 設定=OFF の場合には省略します。
- <binary_block> は、上記 <binary_block> の説明で示したバイナリブロックデータとなります。
- ここに、設定されたデリミタの種類に関わりなく、デリミタは付加されずに、<binary_block> ブロックの最終バイトと同期して EOI 信号を出力します。このため、:REFerence<n>:DATA?クエリは、1 個の＜プログラムメッセージ＞中の最後の＜クエリメッセージユニット＞として下さい。

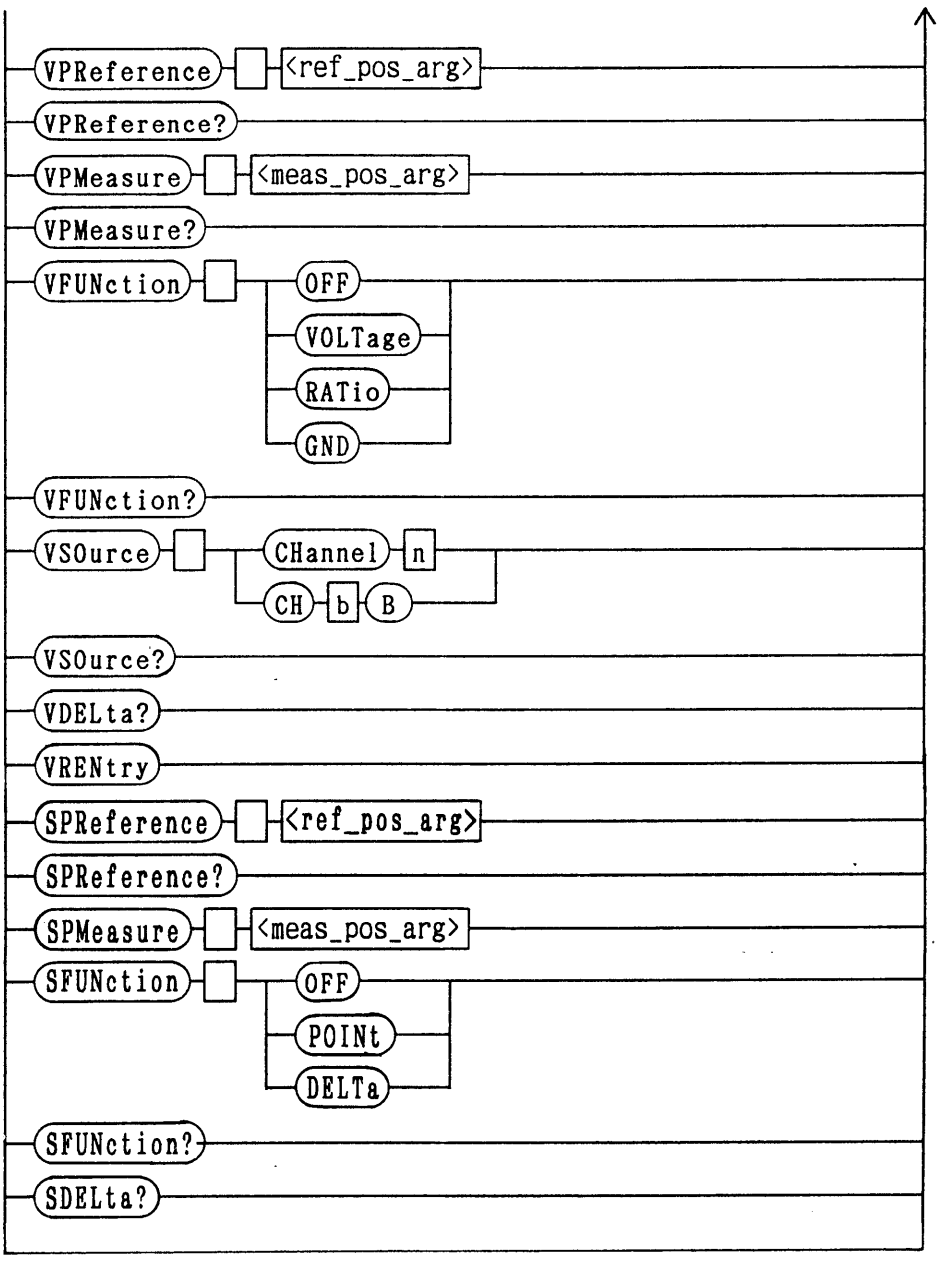
3.10 CURSor 関連コマンド

:CURSor:<item>……　と指定すると、△t/△V カーソル関連の各種設定が可能です。
実際の設定項目は <item> 部のヘッダにより指定されます。

コマンド シンタックス　:CURSor:<item>……
　　ニーモニック　　CURSOR
　　省略形　　CURS

図3.10.1　CURSor関連コマンド全体像




VPReference
<ref_pos_arg>
VPReference?
VPMeasure
<meas_pos_arg>
VPMeasure?
VFUNction
OFF
VOLTage
RATio
GND
VFUNction?
VSOurce
CHannel
n
CH
b
B
VSOurce?
VDELta?
VRENtry
SPReference
<ref_pos_arg>
SPReference?
SPMeasure
<meas_pos_arg>
SFUNction
OFF
POINt
DELTa
SFUNction?
SDELta?

3.10.1 TMOVe コマンド/ TMOVe? クエリ

:CURSor:TMOVe <tmove_arg>

本コマンドで △t カーソルモードを設定します。

コマンドシンタックス　:CURSor:TMOVe <tmove_arg>
　　ニーモニック　TMOVE
　　省略形　TMOV
　　・<tmove_arg> は △t カーソルモード設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
　　・INDependent または LOCK のいずれかとします。

例　PRINT@DSO;":CURSOR:TMOVE LOCK"
　　PRINT@DSO;":CURS:TMOV IND"

:CURSor:TMOVe?

本クエリで現在の △t カーソルモードの設定を読み出します。

クエリシンタックス　:CURSor:TMOVe?
　　ニーモニック　TMOVE?
　　省略形　TMOV?

応答メッセージ　:CURSor:TMOVe <tmove>
　　・「:CURSor:TMOVe」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<tmove> は＜文字応答データ＞フォーマットのニーモニックです。
　　・INDependent または LOCK のいずれかとします。

3.10.2 TPReference コマンド/ TPReference? クエリ

:CURSor:TPReference <ref_pos_arg>

本コマンドで粗い波線で示される △t カーソル(以下リファレンスカーソルと呼びます)の位置を設定します。

エラー処理　　:CURSor:TPReference コマンド/クエリともに、△t カーソルモード(:CURSor:TMOVe)設定=LOCK モードでは無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:CURSor:TPReference <ref_pos_arg>
ニーモニック　TPREFERENCE
省略形　TPR

・<ref_pos_arg> は△t リファレンスカーソル位置設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値とします。
・設定範囲は 0.00～10.00(div)で、内部の設定分解能は 0.02(1divあたり50ﾋﾟｯﾁ)です。本器のグラテキュール(波形表示枠)の左端が 0 に、右端が 10 に相当します。

注　意　　本器は <ref_pos_arg> で与えられた数値を div値とみなしてカーソルを移動しますが、本コマンドはパラメータにサフィックス部を許しません。

例　　PRINT@DSO;":CURSOR:TPREFERENCE 2.00"
PRINT@DSO;":CURS:TPR 2.00E+0"

:CURSor:TPReference?

本クエリで現在の △t リファレンスカーソル位置を読み出します。

クエリシンタックス　:CURSor:TPReference?
ニーモニック　TPREFERENCE?
省略形　TPR?

応答メッセージ　:CURSor:TPReference <ref_pos>

・「:CURSor:TPReference」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<ref_pos> は＜NR3 数値応答データ＞フォーマットで、現在のリファレンスカーソルの位置(div相当)を示す数値です。
・単位を div とみなした数値を送信しますが、サフィックス部は送信しません。

3.10.3 TPRSet コマンド

:CURSor:TPRSet コマンドで△t リファレンスカーソルセットを実行します。

エラー処理　　:CURSor:TPRSet コマンドは、△t カーソルモード(:CURSor:TMOVe)設定＝INDependent モードでは無効で、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:CURSor:TPRSet
ニーモニック　TPRSET
省略形　TPRS

例　PRINT@DSO;":CURSOR:TPRSET"
PRINT@DSO;":CURS:TPRS"

3.10.4 TPMeasure コマンド/ TPMeasure? クエリ

:CURSor:TPMeasure <meas_pos_arg>

本コマンドで細かい波線で示される△t カーソル(以下測定カーソルと呼びます)位置を設定します。

コマンドシンタックス　:CURSor:TPMeasure <meas_pos_arg>
ニーモニック　TPMEASURE
省略形　TPM
・<meas_pos_arg> は △t 測定カーソル位置設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値とします。
・設定範囲は 0.00 ～ 10.00 (div)で、内部の設定分解能は 0.02(1divあたり50ピッチ)です。本器のグラテキュール(波形表示枠)の左端が 0 に、右端が 10 に相当します。

注　意　　本器は <meas_pos_arg> で与えられた数値を div値とみなしてカーソルを移動しますが、本コマンドはパラメータにサフィックス部を許しません。

例　PRINT@DSO;":CURSOR:TPMEASURE 2.00"
PRINT@DSO;":CURS:TPM 2.00E+0"

:CURSor:TPMeasure?

本クエリで、現在の △t 測定カーソル位置を読み出します。

クエリシンタックス　:CURSor:TPMeasure?
ニーモニック　TPMEASURE?
省略形　TPM?

応答メッセージ　:CURSor:TPMeasure <meas_pos>
・「:CURSor:TPMeasure」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<meas_pos> は＜NR3数値応答データ＞フォーマットで、現在の測定カーソルの位置(div相当)を示す数値です。
・単位を div とみなした数値を送信しますが、サフィックス部は送信しません。

3.10.5 TFUNction コマンド/ TFUNction? クエリ

:CURSor:TFUNction <tfunc_arg>

本コマンドで △t 機能の設定をします。

コマンドシンタックス　:CURSor:TFUNction <tfunc_arg>
　　ニーモニック　TFUNCTION
　　省略形　TFUN
　　・<tfunc_arg> は △t 機能選択用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
　　・OFF、TIME、RATio、TRIGger のいずれかとします。

例　PRINT@DSO;":CURSOR:TFUNCTION TIME"
　　PRINT@DSO;":CURS:TFUN TRIG"

:CURSor:TFUNction?

本クエリで、現在の △t 機能の設定を読み出します。

クエリシンタックス　:CURSor:TFUNction?
　　ニーモニック　TFUNCTION?
　　省略形　TFUN?

応答メッセージ　:CURSor:TFUNction <tfunc>
　　・「:CURSor:TFUNction」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<tfunc> は＜文字応答データ＞フォーマットのニーモニックです。
　　・OFF、TIME、RATio、TRIGger のいずれかとなります。

3.10.6 TSOurce コマンド/ TSOurce? クエリ

:CURSor:TSOurce <tsource_arg>

本コマンドで△t 対象時間軸の設定をします。

コマンドシンタックス :CURSor:TSOurce <tsource_arg>
ニーモニック TSOURCE
省略形 TSO
・<tsource_arg> は △t 対象時間軸設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・ASWeep または BSWeep のいずれかとします。

エラー処理 BSWeep 指定はB掃引(:TIMEBASE:BSWeep)設定=ON の場合のみ有効で、それ以外では実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例
```
PRINT@DSO;":CURSOR:TSOURCE ASWEEP"
PRINT@DSO;":CURS:TSO ASW"
```

:CURSor:TSOurce?

本クエリで、現在の △t 対象時間軸を読み出します。

クエリシンタックス :CURSor:TSOurce?
ニーモニック TSOURCE?
省略形 TSO?

応答メッセージ :CURSor:TSOurce <tsource>
・「:CURSor:TSOurce」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<tsource> は＜文字応答データ＞フォーマットのニーモニックです。
・ASWeep または BSWeep のいずれかです。

3.10.7 TRENtry コマンド

:CURSor:TRENtry コマンドで、△t レシオの基準を設定します。

コマンドシンタックス :CURSor:TRENtry
ニーモニック TRENTRY
省略形 TREN

エラー処理 △t 機能(:CURSor:TFUNction)選択≠RATioの場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例
```
PRINT@DSO;":CURSOR:TRENTRY"
PRINT@DSO;":CURS:TREN"
```

3.10.8 TDELta? クエリ

:CURSor:TDELta? クエリで現在の△t カーソル測定結果を読み出します。

クエリシンタックス　:CURSor:TDELta?
　　ニーモニック　TDELTA?
　　省略形　TDEL?

応答メッセージ　<△t_result1>,<△t_result2>
- :CURSor:TDELta? クエリに対する応答データは、:SYSTem:HEADer設定のON/OFF に関わらずヘッダ部を含みません。
- <△t_result1> と <△t_result2> は測定結果を表す数値で、＜NR3 数値応答データ＞フォーマットとします。
- △t 機能(:CURSor:TFUNction)選択=TIME の場合には、<△t_result1>の単位を sec、<△t_result2>の単位を Hz として応答データを作成します。
- △t 機能選択=RATio の場合には、<△t_result1>の単位を PCT(%)、<△t_result2>の単位を DEG として応答データを作成します。
- △t 機能選択=TRIGger の場合には、<△t_result1>、<△t_result2>ともに単位を sec として応答データを作成します。
- △t 機能=OFF などの理由により、正常な測定結果が得られない場合には 9.91E+37 を返します。

注　意　△t 機能選択に関わらずサフィックス部を送信しません。

3.10.9 VPReference コマンド/ VPReference? クエリ

:CURSor:VPReference <ref_pos_arg>

本コマンドで粗い波線で示される△Vカーソル(以下リファレンスカーソルと呼びます)の位置を設定します。

コマンドシンタックス　:CURSor:VPReference <ref_pos_arg>
　　ニーモニック　VPREFERENCE
　　省略形　VPR
・<ref_pos_arg> は△V リファレンスカーソル位置設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値とします。
・設定範囲は -4.00 ～ +4.00 (div)で、内部の設定分解能は 0.02(1divあたり50ピッチ)です。本器のグラテキュール(波形表示枠)の上端が +4 に、下端が -4 に相当します。

注　意　本器は <ref_pos_arg> で与えられた数値を div値とみなしてカーソル移動を実施しますが、本コマンドはパラメータにサフィックス部を許しません。

例
```
PRINT@DSO;":CURSOR:VPREFERENCE 2.00"
PRINT@DSO;":CURS:VPR 2.00E+0"
```

:CURSor:VPReference?

本クエリで、現在の△V リファレンスカーソル位置を読み出します。

クエリシンタックス　:CURSor:VPReference?
　　ニーモニック　VPREFERENCE?
　　省略形　VPR?

応答メッセージ　:CURSor:VPReference <ref_pos>
・「:CURSor:VPReference」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<ref_pos> は＜NR3 数値応答データ＞フォーマットで、現在のリファレンスカーソルの位置(div相当)を示す数値です。
・単位を div とみなした数値を送信しますが、サフィックス部は送信しません。

3.10.10 VPMeasure コマンド/VPMeasure? クエリ

:CURSor:VPMeasure <meas_pos_arg>

本コマンドで細かい波線で示される△V カーソル(以下測定カーソルと呼びます)位置を設定します。

コマンドシンタックス　:CURSor:VPMeasure <meas_pos_arg>
　　ニーモニック　VPMEASURE
　　省略形　VPM

・<meas_pos_arg> は △V 測定カーソル位置設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値とします。
・設定範囲は -4.00 ～ +4.00 (div)で、内部の設定分解能は 0.02(1divあたり 50 ピッチ)です。本器のグラテキュール(波形表示枠)の上端が +4 に、下端が -4 に相当します。

注　意　本器は <meas_pos_arg> で与えられた数値を div値とみなしてカーソルを移動しますが、本コマンドはパラメータにサフィックス部を許しません。

例
```
PRINT@DSO;":CURSOR:VPMEASURE 2.00"
PRINT@DSO;":CURS:VPM 2.00E+0"
```

:CURSor:VPMeasure?

本クエリで、現在の △V 測定カーソル位置を読み出します。

クエリシンタックス　:CURSor:VPMeasure?
　　ニーモニック　VPMEASURE?
　　省略形　VPM?

応答メッセージ　:CURSor:VPMeasure <meas_pos>

・「:CURSor:VPMeasure」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<meas_pos> は＜NR3数値応答データ＞フォーマットで、現在の△V 測定カーソルの位置(div相当)を示す数値です。
・単位を div とみなした数値を送信しますが、サフィックス部は送信しません。

3.10.11 VFUNction コマンド/ VFUNction? クエリ

:CURSor:VFUNction <vfunc_arg>

本コマンドで△V を設定します。

コマンドシンタックス :CURSor:VFUNction <vfunc_arg>
ニーモニック VFUNCTION
省略形 VFUN
・<vfunc_arg> は△V 機能選択用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・OFF、VOLTage、RATio、GND のいずれかとします。

例
```
PRINT@DSO;":CURSOR:VFUNCTION VOLTAGE"
PRINT@DSO;":CURS:VFUN VOLT"
```

:CURSor:VFUNction?

本クエリで現在の △V 機能の設定を読み出します。

クエリシンタックス :CURSor:VFUNction?
ニーモニック VFUNCTION?
省略形 VFUN?

応答メッセージ :CURSor:VFUNction <vfunc>
・「:CURSor:VFUNction」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<vfunc> は＜文字応答データ＞フォーマットのニーモニックです。
・OFF、VOLTage、RATio、GND のいずれかとします。

3.10.12 VSOurce コマンド/ VSOurce? クエリ

:CURSor:VSOurce <vsource_arg>

本コマンドで△V 対象チャネルの設定をします。

コマンドシンタックス　:CURSor:VSOurce <vsource_arg>
　　ニーモニック　VSOURCE
　　省略形　VSO
・<vsource_arg> は△V 対象チャネル設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B のいずれかとします。

エラー処理　CH1B、CH2B 指定は、B掃引(:TIMEBASE:BSWeep)設定=OFF の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例
```
PRINT@DSO;":CURSOR:VSOURCE CHANNEL1"
PRINT@DSO;":CURS:VSO CH2"
```

:CURSor:VSOurce?

本クエリで、現在の △V 対象チャネル設定を読み出します。

クエリシンタックス　:CURSor:VSOurce?
　　ニーモニック　VSOURCE?
　　省略形　VSO?

応答メッセージ　:CURSor:VSOurce <vsource>
・「:CURSor:VSOurce」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<vsource> は＜文字応答データ＞フォーマットのニーモニックです。
・CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B のいずれかとします。

3.10.13 VRENtry コマンド

:CURSor:VRENtry コマンドで △V レシオの基準を設定します。

エラー処理　　△V 機能(:CURSor:VFUNction)選択≠RATio の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

コマンドシンタックス　:CURSor:VRENtry
ニーモニック　VRENTRY
省略形　VREN

例　　PRINT@DSO;":CURSOR:VRENTRY"
PRINT@DSO;":CURS:VREN"

3.10.14 VDELta? クエリ

:CURSor:VDELta? クエリで現在の△V カーソル測定結果を読み出します。

クエリシンタックス　:CURSor:VDELta?
ニーモニック　VDELTA?
省略形　VDEL?

応答メッセージ　<△v_result1>,<△v_result2>
・:CURSor:VDELta? クエリに対する応答データは、:SYSTem:HEADer 設定の ON/OFF に関わらずヘッダ部を含みません。
・<△V_result1> と <△V_result2> は測定結果を表す数値で、＜NR3 数値応答データ＞フォーマットとします。
・△V 機能(:CURSor:VFUNction)選択=VOLTage の場合には <△V_result1> の単位を V として応答データを作成します。<△V_result2>=9.91E+37 となります。
・△V 機能選択=RATio の場合には、<△V_result1>の単位を PCT(%)、<△V_result2>の単位を dB として応答データを作成します。
・△V 機能選択=GND の場合には、<△V_result1>、<△V_result2>ともに単位を V として応答データを作成します。
・△V 機能=OFF などの理由により、正常な測定結果が得られない場合には 9.91E+37 を返します。

注　意　　△V 機能選択に関わらずサフィックス部を送信しません。

3.10.15 SPReference コマンド/ SPReference? クエリ

:CURSor:SPReference <ref_pos_arg>

本コマンドで粗い波線で示される スペクトルリファレンスカーソル（以下リファレンスカーソルと呼びます）の位置を設定します。

コマンドシンタックス　:CURSor:SPReference <ref_pos_arg>
　　ニーモニック　SPREFERENCE
　　省略形　SPR
・<ref_pos_arg> はリファレンスカーソル位置設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値とします。
・設定範囲は 0.00～10.00 (div) で、内部の設定分解能は 0.02 (1div あたり 50 ﾋﾟｯﾁ) です。本器のグラテキュール（波形表示枠）の左端が 0 に、右端が 10 に相当します。

注　意　本器は <ref_pos_arg> で与えられた数値を div値とみなしてカーソルを移動しますが、本コマンドはパラメータにサフィックス部を許しません。

例
```
PRINT@DSO;":CURSOR:SPREFERENCE 2.00"
PRINT@DSO;":CURS:SPR 2.00E+0"
```

:CURSor:SPReference?

本クエリで現在の スペクトルリファレンスカーソル位置を読み出します。

クエリシンタックス　:CURSor:SPReference?
　　ニーモニック　SPREFERENCE?
　　省略形　SPR?

応答メッセージ　:CURSor:SPReference <ref_pos>
・「:CURSor:SPReference」部分は :SYSTem:HEADer 設定=OFF の場合には省略します。
・<ref_pos> は＜NR3 数値応答データ＞フォーマットで、現在のリファレンスカーソルの位置（div 相当）を示す数値です。
・単位を div とみなした数値を送信しますが、サフィックス部は送信しません。

3.10.16 SPMeasure コマンド/ SPMeasure? クエリ

:CURSor:SPMeasure <meas_pos_arg>

本コマンドで細かい波線で示されるスペクトル測定カーソル（以下測定カーソルと呼びます）位置を設定します。

コマンドシンタックス　:CURSor:SPMeasure <meas_pos_arg>
　　ニーモニック　SPMEASURE
　　省略形　SPM
- <meas_pos_arg> は測定カーソル位置設定用のパラメータで、＜数値プログラムデータ＞フォーマットの数値とします。
- 設定範囲は 0.00～10.00（div）で、内部の設定分解能は 0.02（1div あたり 50 ﾋﾟｯﾁ）です。本器のグラテキュール（波形表示枠）の左端が 0 に、右端が 10 に相当します。

注　意　本器は <meas_pos_arg> で与えられた数値を div 値とみなしてカーソルを移動しますが、本コマンドはパラメータにサフィックス部を許しません。

例
```
PRINT@DSO;":CURSOR:SPMEASURE 2.00"
PRINT@DSO;":CURS:SPM 2.00E+0"
```

:CURSor:SPMeasure?

本クエリで現在のスペクトル測定カーソル位置を読み出します。

クエリシンタックス　:CURSor:SPMeasure?
　　ニーモニック　SPMEASURE?
　　省略形　SPM?

応答メッセージ　:CURSor:SPMeasure <meas_pos>
- 「:CURSor:SPMeasure」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
- <meas_pos> は＜NR3 数値応答データ＞フォーマットで、現在の測定カーソルの位置（div 相当）を示す数値です。
- 単位を div とみなした数値を送信しますが、サフィックス部は送信しません。

3.10.17 SFUNction コマンド/ SFUNction? クエリ

:CURSor:SFUNction <sfunc_arg>

本コマンドでスペクトル機能の設定をします。

コマンドシンタックス　:CURSor:SFUNction <sfunc_arg>
　　ニーモニック　SFUNCTION
　　省略形　SFUN
　　・<sfunc_arg> はスペクトル機能選択用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
　　・OFF、POINt、DELTa のいずれかとします。POINt または DELTa を設定すると、Δt および ΔV 機能が OFF のなります。

例　PRINT@DSO;":CURSOR:SFUNCTION POINT"
　　PRINT@DSO;":CURS:SFUN DELT"

:CURSor:SFUNction?

本クエリで現在のスペクトル機能の設定を読み出します。

クエリシンタックス　:CURSor:SFUNction?
　　ニーモニック　SFUNCTION?
　　省略形　SFUN?

応答メッセージ　:CURSor:SFUNction <sfunc>
　　・「:CURSor:SFUNction」部分は :SYSTem:HEADer 設定=OFF の場合には省略します。
　　・<sfunc> は＜文字応答データ＞フォーマットのニーモニックです。
　　・OFF、POINt、DELTa のいずれかとなります。

3.10.18 SDELta? クエリ

:CURSor:SDELta? クエリで現在のスペクトルカーソル測定結果を読み出します。

クエリシンタックス　:CURSor:SDELta?
　　ニーモニック　SDELTA?
　　省略形　SDEL?

応答メッセージ　<s_result1>,<s_result2>,<s_result3>,<s_result4>
- :CURSor:SDELta? クエリに対する応答データは :SYSTem:HEADer 設定の ON/OFF に関わらずヘッダ部を含みません。
- <s_result1>,<s_result2>,<s_result3> および <s_result4> は測定結果を表す数値で、＜NR3 数値応答データ＞フォーマットとします。
- スペクトル機能 (:CURSor:SFUNction) 選択=POINt の場合は、
  <s_result1>はリファレンスカーソルの dBV 測定結果で単位は dBV です。
  <s_result2>はリファレンスカーソルの周波数測定結果で単位は Hz です。
  <s_result3>は測定カーソルの dBV 測定結果で単位は dBV です。
  <s_result4>は測定カーソルの周波数測定結果で単位は Hz です。
- スペクトル機能 (:CURSor:SFUNction) 選択=DELTa の場合は、
  <s_result1>はカーソル間の dBV 差測定結果で単位は dBV です。
  <s_result2>はカーソル間の周波数差測定結果で単位は Hz です。
  <s_result3>と<s_result4>は 9.91E+37 を返します。
- スペクトル機能=OFF などの理由により、正常な測定結果が得られない場合には 9.91E+37 を返します。

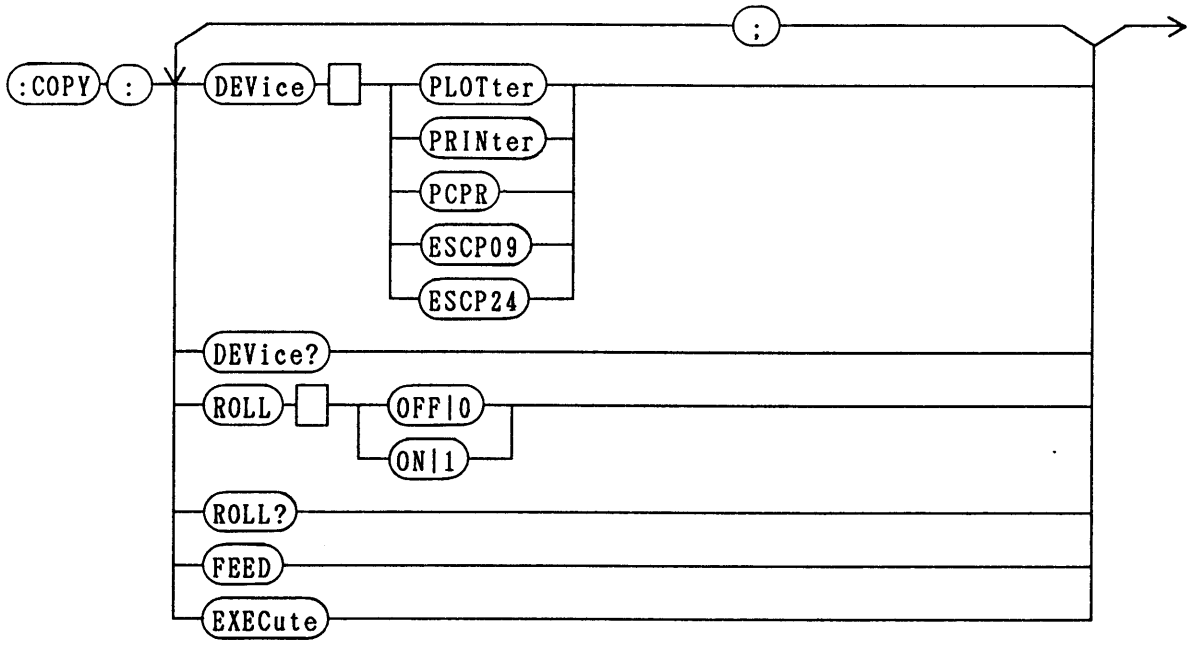
注　意　スペクトル機能選択に関わらずサフィックス部を送信しません。

3.11 COPY コマンド

:COPY:<item>……　と指定すると、コピー機能を対象とする各種設定および動作の実施が可能です。
実際の設定項目は <item> 部のヘッダにより指定されます。
なお、プロッタサブメニューに関する項目は PLOT 親コマンド下に独立させます。

コマンドシンタックス :COPY:<item>……
　　ニーモニック COPY
　　省略形　　COPY(省略形なし)

図3.11.1 COPY 関連コマンド全体像



3.11.1 DEViceコマンド/ DEVice? クエリ

:COPY:DEVice <device_arg>

本コマンドでコピーデバイス(と外部プリンタ用制御コード)を設定します。

コマンドシンタックス　:COPY:DEVice <device_arg>
　　ニーモニック　DEVICE
　　省略形　DEV

・<device_arg> はコピーデバイスの種類設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・PLOTter、PRINter、PCPR、ESCP09、ESCP24 のいずれかとします。
・ESCP09 と ESCP24 の先頭 4 文字が互いにユニークではないため、この 2 つには省略形を許しません。
・PCPR、ESCP09、ESCP24 のいずれかが指定された場合、コピーデバイスを PARALLEL とし、指定されたパラメータに応じて制御コードを設定します。

エラー処理

・内蔵プリンタ(ｵﾌﾟｼｮﾝ)が実装されていない場合、PRINTer 指定はコマンドエラーとします。
・RS & PARALLEL I/F(ｵﾌﾟｼｮﾝ)が実装されていない場合、PCPR、ESCP09、ESCP24 指定はコマンドエラーとします。
・いずれの場合も、以下のメッセージをエラーキューに格納します。
　-141,"Invalid character data"

例

```
PRINT@DSO;":COPY:DEVICE PLOTTER"
PRINT@DSO;":COPY:DEV PRIN"
```

:COPY:DEVice?

本クエリで、現在のコピーデバイスの設定を読み出します。

クエリシンタックス　:COPY:DEVice?
　　ニーモニック　DEVICE?
　　省略形　DEV?

応答メッセージ　:COPY:DEVice <device>

・「:COPY:DEVice」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<device> は＜文字応答データ＞フォーマットのニーモニックです。
・PLOTter、PRINter、PCPR、ESCP09、ESCP24 のいずれかとします。

3.11.2 ROLL コマンド/ ROLL? クエリ

:COPY:ROLL <mode_arg>

本コマンドで ロールプリント(ON/OFF)を設定します。

コマンドシンタックス　:COPY:ROLL <mode_arg>
ニーモニック　ROLL
省略形　ROLL(省略形なし)

・<mode_arg>は ON/OFF 設定用のパラメータで、<文字プログラムデータ>フォーマットの ON または OFF、<数値プログラムデータ>フォーマットの数値のいずれかとします。
・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

エラー処理

・:COPY:ROLL コマンド/クエリともに、内蔵プリンタ(ｵﾌﾟｼｮﾝ)が実装されていない場合、コマンドエラーとし、エラーキューに以下のメッセージを格納します。
-113,"Undefined header"
・ON の指定はメモリ使用種別(:ACQuire:MEMory 設定)がロール(ROLL)でかつ、ロールモード(:ACQuire:RMODe 設定)がページドロール(PAGed)の場合は実行エラーとし、エラーキューに以下のメッセージを格納します。
-221,"Setting conflict"

:COPY:ROLL?

本クエリで 現在のロールプリントの設定(ON/OFF)を読み出します。

クエリシンタックス　COPY:ROLL?
ニーモニック　ROLL?
省略形　ROLL?(省略形なし)

応答メッセージ　:COPY:ROLL <mode>

・「:COPY:ROLL」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode>は<NR1 数値応答データ>フォーマットとし、0 または 1 の 1 桁の数字とします。
・0 が OFF、1 が ON を意味します。

3.11.3 FEED コマンド

:COPY:FEED コマンドで紙送りを実行します。

コマンドシンタックス :COPY:FEED
　　ニーモニック　FEED
　　省略形　　FEED(省略形なし)

エラー処理　　:COPY:DEVice設定=PRINter の状態で、FEED サブコマンドを実行しようとして、ペーパーエンプティなどのエラーが内蔵プリンタユニットから通知された場合、実行エラーとし、以下のメッセージをエラーキューに格納します。
　-240,"Hardware error"

例　　PRINT@DSO;":COPY:FEED"

3.11.4 EXECute コマンド

:COPY:EXECute コマンドでコピーを実行します。

コマンドシンタックス :COPY:EXECute
　　ニーモニック EXECute
　　省略形　　EXEC

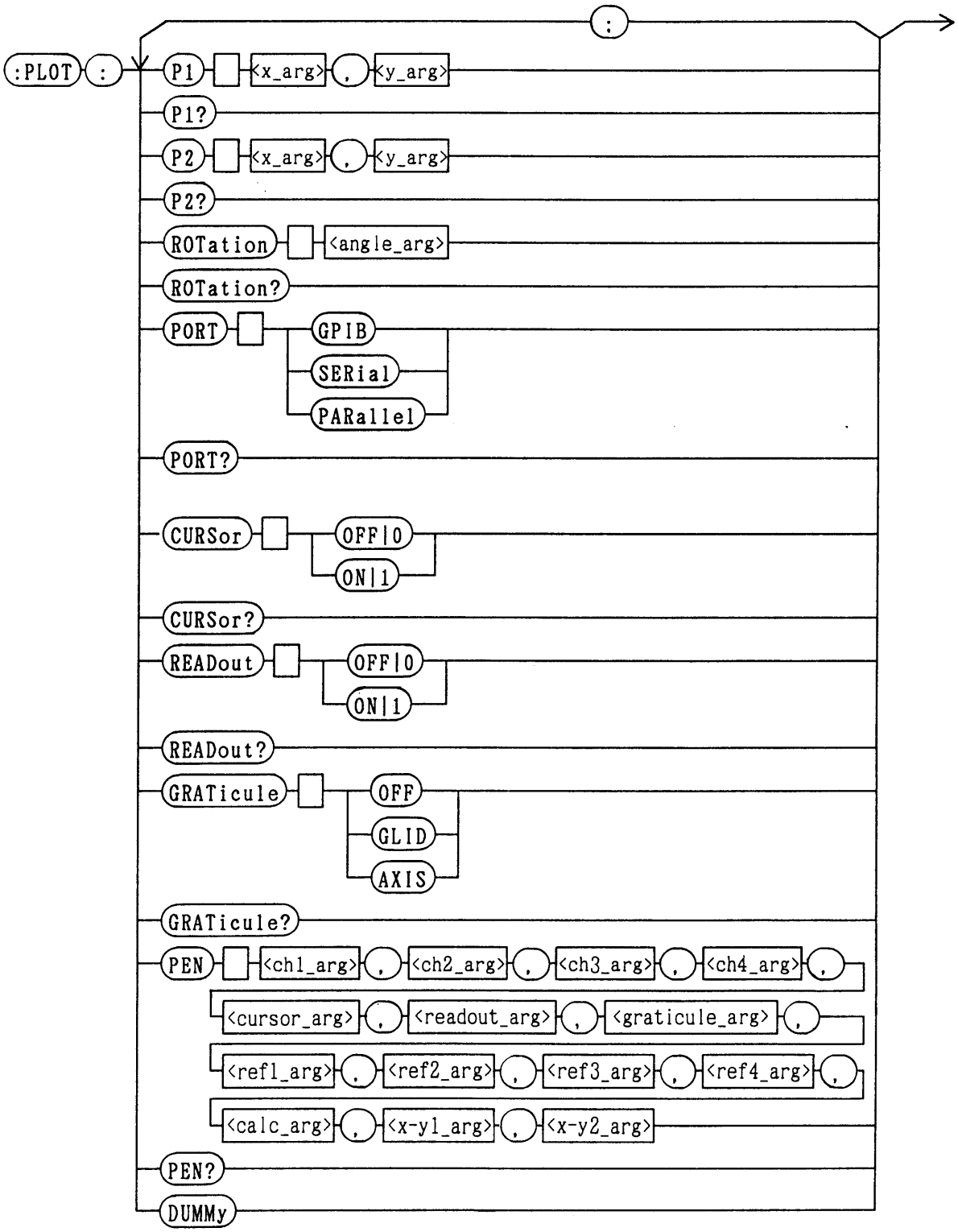
エラー処理　　:COPY:DEVice設定=PRINter の状態で、COPY サブコマンドを実行しようとして、ペーパーエンプティなどのエラーが内蔵プリンタユニットから通知された場合、実行エラーとし、以下のメッセージをエラーキューに格納します。
　-240,"Hardware error"

例　　PRINT@DSO;":COPY:EXECUTE"
　　PRINT@DSO;":COPY:EXEC"

3.12 PLOT コマンド

:PLOT:<item>……　と指定すると、プロッタ出力関連の各種設定および動作の実施が可能です。
実際の設定項目は <item> 部のヘッダにより指定されます。

コマンドシンタックス　:PLOT:<item>……
　　ニーモニック　　PLOT
　　省略形　　PLOT(省略形なし)



3.12.1 P1 コマンド/ P1? クエリ

:PLOT:P1 <x_arg>,<y_arg>

本コマンドで P1 位置を設定します。

コマンドシンタックス　:PLOT:P1 <x_arg>,<y_arg>
　　ニーモニック　P1
　　省略形　P1(省略形なし)
　　・<x_arg>,<y_arg> は P1 位置設定用のパラメータで、いずれも＜数値プログラムデータ＞フォーマットの数値です。
　　・単位は mm とみなしますが、サフィックス部を許しませんので、数値のみ送信して下さい。

例　PRINT@DSO;":PLOT:P1 100,100"
　　PRINT@DSO;":PLOT:P1 1E+2"

:PLOT:P1?

本クエリで現在の P1 位置の設定値を読み出します。

クエリシンタックス　:PLOT:P1?
　　ニーモニック　P1?
　　省略形　P1?(省略形なし)

応答メッセージ　:PLOT:P1 <x>,<y>
　　・「:PLOT:P1」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<x>,<y> ともに＜NR1 数値応答データ＞の数値とします。
　　・単位は mm とみなして応答データを作成しますが、サフィックス部は送信しません。

3.12.2 P2 コマンド/ P2? クエリ

:PLOT:P2 <x_arg>,<y_arg>

本コマンドで P2 位置の設定をします。

コマンドシンタックス　:PLOT:P2 <x_arg>,<y_arg>
ニーモニック　P2
省略形　P2(省略形なし)
・<x_arg>,<y_arg> は P2 位置設定用のパラメータで、いずれも＜数値プログラムデータ＞フォーマットとします。
・単位は mm とみなしますが、サフィックス部を許しませんので、数値のみ送信して下さい。

例
```
PRINT@DSO;":PLOT:P2 100,100"
PRINT@DSO;":PLOT:P2 1E+2"
```

:PLOT:P2?

本クエリで、現在の P2 位置の設定値を読み出します。

クエリシンタックス　:PLOT:P2?
ニーモニック　P2?
省略形　P2?(省略形なし)

応答メッセージ　:PLOT:P2 <x>,<y>
・「:PLOT:P2」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<x>,<y> ともに＜NR1 数値応答データ＞フォーマットの数値です。
・単位は mm とみなして応答データを作成しますが、サフィックス部は送信しません。

3.12.3 ROTation コマンド/ ROTation? クエリ

:PLOT:ROTation <angle_arg>

本コマンドでプロッタで作画する場合のローテーションを設定します。

コマンドシンタックス　:PLOT:ROTation <angle_arg>
　　ニーモニック　ROTATION
　　省略形　ROT
・<angle_arg> はローテーション設定用のパラメータで、<数値プログラムデータ>フォーマットを基本とし、サフィックス部も有効とします。
・デフォルトのサフィックス乗数は$\times 10^{0}$、デフォルトのサフィックス単位は DEG とします。
・設定値は丸めにより 0、90、180、270 のいずれか近い値に飛びつかせます。

エラー処理　<angle_arg> のサフィックス単位としては DEG のみ有効で、これ以外の文字を受け取った場合はコマンドエラーとし、以下のメッセージをエラーキューに格納します。
-131,"Invalid suffix"

例
```
PRINT@DSO;":PLOT:ROTATION 0"
PRINT@DSO;":PLOT:ROT 270"
```

:PLOT:ROTation?

本クエリで現在のローテーションの設定値を読み出します。

クエリシンタックス　:PLOT:ROTation?
　　ニーモニック　ROTATION?
　　省略形　ROT?

応答メッセージ　:PLOT:ROTation <angle>
・「:PLOT:ROTation」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<angle> は<NR1 数値応答データ>の数値です。
・0、90、180、270 のいずれかとなります。
・単位は DEG とみなしますが、サフィックス部は送信しません。

3.12.4 PORT コマンド/ PORT? クエリ

:PLOT:PORT <port_arg>

本コマンドでプロッタインタフェースの設定をします。

コマンドシンタックス　:PLOT:PORT <port_arg>
　　ニーモニック　PORT
　　省略形　PORT(省略形なし)
・<port_arg> はプロッタインタフェースの種類設定用のパラメータで、<文字プログラムデータ>フォーマットのニーモニックです。
・GPIB、SERial、PARallel のいずれかとします。

エラー処理
・RS & PARALLEL I/F(オプション)が実装されていない場合、SERial または PARallel 指定はコマンドエラーとし、以下のメッセージをエラーキューに格納します。
　-113,"Undefined header"
・GP-IB インタフェース経由で :PLOT:PORT設定=GPIB が指定された場合、RS-232C インタフェース経由で :PLOT:PORT設定=SERial が指定された場合は実行エラーとし、以下のメッセージをエラーキューに格納します。
　-221,"Setting conflict"

例
```
PRINT@DSO;":PLOT:PORT SERial"
PRINT@DSO;":PLOT:PORT PAR"
```

:PLOT:PORT?

本クエリで、現在のプロッタインタフェースの設定を読み出します。

クエリシンタックス　:PLOT:PORT?
　　ニーモニック　PORT?
　　省略形　PORT?(省略形なし)

応答メッセージ　:PLOT:PORT <port>
・「:PLOT:PORT」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<port> は<文字応答データ>フォーマットのニーモニックです。
・GPIB、SERial、PARallel のいずれかとします。

3.12.5 CURSor コマンド/ CURSor? クエリ

:PLOT:CURSor <mode_arg>

本コマンドでカーソルプロットを設定(ON/OFF)します。

コマンドシンタックス　:PLOT:CURSor <mode_arg>
　　ニーモニック　CURSOR
　　省略形　CURS
　　・<mode_arg> は ON/OFF 設定用のパラメータで、＜文字プログラムデータ＞フォーマットの ON または OFF、＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
　　・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

例　PRINT@DSO;":PLOT:CURSOR ON"
　　PRINT@DSO;":PLOT:CURS 1"

:PLOT:CURSor?

本クエリで、現在のカーソルプロットの設定(ON/OFF)を読み出します。

クエリシンタックス　:PLOT:CURSor?
　　ニーモニック　CURSOR?
　　省略形　CURS?

応答メッセージ　:PLOT:CURSor <mode>
　　・「:PLOT:CURSor」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の 1 桁の数字とします。
　　・0 が OFF、1 が ON を意味します。

3.12.6 READout コマンド/ READout? クエリ

:PLOT:READout <mode_arg>

本コマンドでリードアウトプロットの設定(ON/OFF)をします。

コマンドシンタックス :PLOT:READout <mode_arg>
ニーモニック READOUT
省略形 READ

・<mode_arg> は ON/OFF 設定用のパラメータで、＜文字プログラムデータ＞フォーマットの ON または OFF、＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

例
```
PRINT@DSO;":PLOT:READOUT ON"
PRINT@DSO;":PLOT:READ 1"
```

:PLOT:READout?

本クエリで、現在のリードアウトプロットの設定(ON/OFF)を読み出します。

クエリシンタックス :PLOT:READout?
ニーモニック READOUT?
省略形 READ?

応答メッセージ :PLOT:READout <mode>

・「:PLOT:READout」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットの数値で、0 または 1 の 1 桁の数字とします。
・0 が OFF、1 が ON を意味します。

3.12.7 GRATicule コマンド/ GRATicule? クエリ

:PLOT:GRATicule <graticule_arg>

本コマンドでプロットするグラテキュールの種類の設定をします。

コマンドシンタックス　:PLOT:GRATicule <graticule_arg>
　　ニーモニック　GRATICULE
　　省略形　GRAT
　　・<graticule_arg> はプロットするグラテキュールの種類設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
　　・OFF、GLID、AXIS のいずれかとします。

例　PRINT@DSO;":PLOT:GRATICULE GLID"
　　PRINT@DSO;":PLOT:GRAT OFF"

:PLOT:GRATicule?

本クエリで、現在のプロットするグラテキュールの設定を読み出します。

クエリシンタックス　:PLOT:GRATicule?
　　ニーモニック　GRATICULE?
　　省略形　GRAT?

応答メッセージ　:PLOT:GRATicule <graticule>
　　・「:PLOT:GRATicule」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<graticule> は＜文字応答データ＞フォーマットのニーモニックです。
　　・OFF、GLID、AXIS のいずれかとします。

3.12.8 PEN コマンド/ PEN? クエリ

:PLOT:PEN <ch1_arg>,<ch2_arg>,…,<x-y2_arg>

本コマンドでペン番号を設定します。

コマンドシンタックス :PLOT:PEN <ch1_arg>,<ch2_arg>,<ch3_arg>,<ch4_arg>,
<curs_arg>,<read_arg>,<grat_arg>,<ref1_arg>,
<ref2_arg>,<ref3_arg>,<ref4_arg>,<calc_arg>,
<x-y1_arg>,<x-y2_arg>

ニーモニック PEN

省略形 PEN(省略形なし)

<ch1_arg>,<ch2_arg>,…,<x-y2_arg> はペン番号設定用のパラメータで、いずれも＜数値プログラムデータ＞フォーマットの数値とします。

- <ch1_arg> CH1 作画用ペン番号
- <ch2_arg> CH2 作画用ペン番号
- <ch3_arg> CH3 作画用ペン番号
- <ch4_arg> CH4 作画用ペン番号
- <curs_arg> カーソル作画用ペン番号
- <read_arg> リードアウト作画用ペン番号
- <grat_arg> グラテキュール作画用ペン番号
- <ref1_arg> リファレンス波形 1 作画用ペン番号
- <ref2_arg> リファレンス波形 2 作画用ペン番号
- <ref3_arg> リファレンス波形 3 作画用ペン番号
- <ref4_arg> リファレンス波形 4 作画用ペン番号
- <calc_arg> 演算波形作画用ペン番号
- <x-y1_arg> X-Y1 波形作画用ペン番号
- <x-y2_arg> X-Y2 波形作画用ペン番号

例

```
PRINT@DSO;":PLOT:PEN 1,2,3,4,5,6"
PRINT@DSO;":PLOT:PEN 1,,,,,6"
```

:PLOT:PEN?

本クエリで現在のペン番号の設定値を読み出します。

クエリシンタックス :PLOT:PEN?

ニーモニック PEN?

省略形 PEN?(省略形なし)

応答メッセージ :PLOT:PEN <ch1>,<ch2>,<ch3>,<ch4>,<curs>,<read>,<grat>,
<ref1>,<ref2>,<ref3>,<ref4>,<calc>,<x-y1>,<x-y2>

・「:PLOT:PEN」部分は :SYSTem:HEADer設定=OFF の場合には省略します。

・パラメータは総て＜NR1 数値応答データ＞フォーマットの数値とします。

3.12.9 DUMMy コマンド

:PLOT:DUMMy コマンドでダミープロットを実行します。

コマンドシンタックス :PLOT:DUMMy

ニーモニック DUMMY

省略形 DUMM

例

```
PRINT@DSO;":PLOT:DUMMY"
PRINT@DSO;":PLOT:DUMM"
```

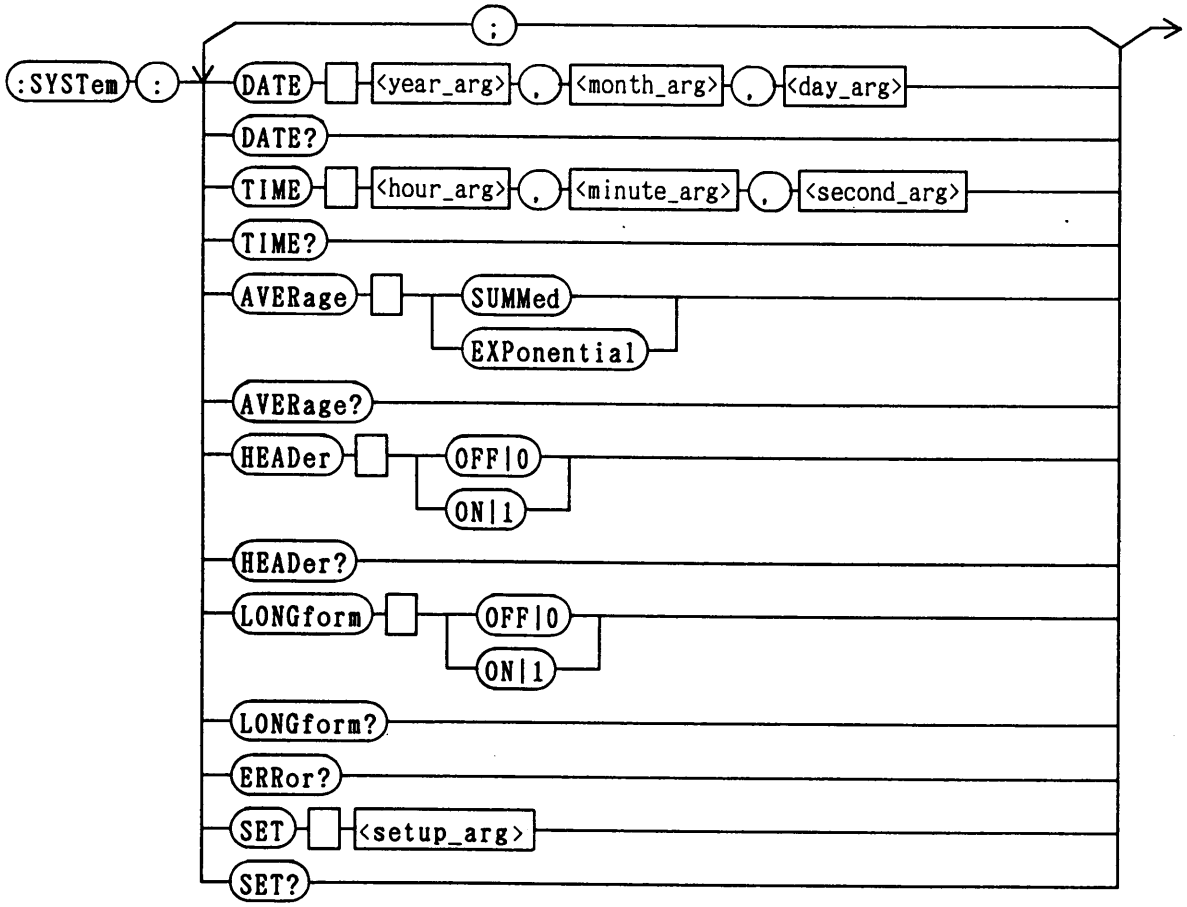
3.13 SYSTem 関連コマンド

:SYSTem:<item>…… と指定すると、システム全体を対象とする各種設定が可能です。
実際の設定項目は <item> 部のヘッダにより指定されます。
なお、ローカル操作時にシステムメニュー画面で設定される項目の中で、以下の項目については：
SYSTem 親コマンド下に含めません。

日付表示	→ :DISPlay 親コマンド参照
オートセットアップモード	→ ルートレベルコマンド参照
△t カーソルモード	→ :CURSor 親コマンド参照
RGB出力	→ :DISPlay 親コマンド参照
リモートインタフェース	→ リモート制御できません
自動校正とテスト	→ リモート制御できません

コマンドシンタックス :SYSTem:<item>……
ニーモニック SYSTEM
省略形 SYST

図3.13.1 SYSTem 関連コマンド全体像



3.13.1 DATE コマンド/ DATE? クエリ

:SYSTem:DATE <year_arg>,<month_arg>,<day_arg>

本コマンドで本器の内部時計の日付を設定します。

コマンドシンタックス　:SYSTem:DATE <year_arg>,<month_arg>,<day_arg>
　　ニーモニック　DATE
　　省略形　DATE(省略形なし)
・<year_arg> は年号、<month_arg> は月、<day_arg> は日付設定用のパラメータで、何れも＜数値プログラムデータ＞フォーマットです。
・設定値は、丸めを行ってから整数値に変換して解釈します。
・<year_arg> の設定可能範囲は 1980～2079 です。
・<month_arg> の設定可能範囲は 1～12 です。
・<day_arg> の設定可能範囲は 1～31 です。

エラー処理
・範囲外の数値を指定された場合、実行エラーとします。
・<year_arg> と <month_arg> と <day_arg> は対をなすパラメータであり、年月日として許されない組み合わせになった場合も実行エラーとします。
・いずれの場合も、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例
```
PRINT@DSO;":SYSTEM:DATE 1991,8,2"
PRINT@DSO;":SYST:DATE 1991,08,02"
PRINT@DSO;":SYST:DATE ,08,02"
PRINT@DSO;":SYST:DATE ,,31"
                        …… 2/4/6/9/11 月は実行エラーとなる。
PRINT@DSO;":SYST:DATE ,2,29"
                            …… 閏年以外は実行エラーとなる。
```

:SYSTem:DATE?

本クエリで、現在の内部時計から年月日を読み出します。

クエリシンタックス　:SYSTem:DATE?
　　ニーモニック　DATE?
　　省略形　DATE?(省略形なし)

応答メッセージ　:SYSTem:DATE <year>,<month>,<day>
・「:SYSTem:DATE」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<year> は年号、<month> は月、<day> は日を表す数値です。
何れも＜NR1 数値応答データ＞フォーマットとします。

3.13.2 TIME コマンド/ TIME? クエリ(SCPI)

:SYSTem:TIME <hour_arg>,<minute_arg>,<second_arg>

本コマンドで本器の内部時計の時刻を設定します。

コマンドシンタックス　:SYSTem:TIME <hour_arg>,<minute_arg>,<second_arg>
　　ニーモニック　TIME
　　省略形　TIME(省略形なし)
・<hour_arg> は時、<minute_arg> は分、<second_arg> は秒設定用のパラメータで、何れも＜数値プログラムデータ＞フォーマットです。
・設定値は、丸めを行って整数値に変換して解釈します。
・<hour_arg> の設定可能範囲は 0～23 です。
・<minute_arg> の設定可能範囲は 0～59 です。
・<second_arg> の設定可能範囲は 0～59 です。

エラー処理　範囲外の数値を指定された場合、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例
```
PRINT@DSO;":SYSTEM:TIME 23,59,59"
PRINT@DSO;":SYST:TIME 0,0,0"
PRINT@DSO;":SYST:TIME 12,00"
PRINT@DSO;":SYST:TIME ,00,00"
PRINT@DSO;":SYST:TIME ,,00"
```

:SYSTem:TIME?

本クエリで内部時計の時刻を読み出します。

クエリシンタックス　:SYSTem:TIME?
　　ニーモニック　TIME?
　　省略形　TIME?(省略形なし)

応答メッセージ　:SYSTem:TIME <hour>,<minute>,<second>
・「:SYSTem:TIME」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<hour> は時、<minute> は分、<second> は秒を表す数値です。
何れも＜NR1 数値応答データ＞フォーマットとします。

3.13.3 HEADer コマンド/ HEADer? クエリ

:SYSTem:HEADer <mode_arg>

本コマンドで本器がリモートインタフェースを介して送信する応答メッセージにヘッダ部を付与するか否かの設定をします。なお、クエリの種類によっては、:SYSTem:HEADer設定に関わらず、応答メッセージにヘッダ部を含むか否かが固定的に定まっていますので、使用するクエリ毎に応答メッセージのフォーマットをご確認下さい。

コマンドシンタックス　:SYSTem:HEADer <mode_arg>
　　ニーモニック　HEADER
　　省略形　HEAD

・<mode_arg>は ON/OFF 設定用のパラメータで、＜文字プログラムデータ＞フォーマットの ON または OFF、＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

例

```
PRINT@DSO;":SYSTEM:HEADER ON"
PRINT@DSO;":SYST:HEAD 1"
```

:SYSTem:HEADer?

本クエリで応答メッセージにヘッダ部を付与するか否かの設定を読み出します。

クエリシンタックス　:SYSTem:HEADer?
　　ニーモニック　HEADER?
　　省略形　HEAD?

応答メッセージ　:SYSTem:HEADer <mode>

・「:SYSTem:HEADer」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の1桁の数字とします。
・0 が OFF、1 が ON を意味します。

3.13.4 LONGform コマンド/ LONGform? クエリ

:SYSTem:LONGform <mode_arg>

本コマンドで本器がリモートインタフェースを介して送信する＜応答メッセージ＞にヘッダ部を付与する場合にフルスペルとするか省略形とするかを設定します。
また、＜文字応答データ＞をフルスペルとするか否かも、この設定により決定されます。

コマンドシンタックス　:SYSTem:LONGform <mode_arg>
　　ニーモニック　LONGFORM
　　省略形　LONG
・<mode_arg> は ON/OFF 設定用のパラメータで、＜文字プログラムデータ＞フォーマットの ON または OFF、＜数値プログラムデータ＞フォーマットの数値のいずれかとします。
・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

例　PRINT@DSO;":SYSTEM:LONGFORM ON"
　　PRINT@DSO;":SYST:LONG 1"

:SYSTem:LONGform?

本クエリで＜応答メッセージ＞のヘッダ部と＜文字応答データ＞のニーモニックがフルスペルか省略形かの設定状態を読み出します。

クエリシンタックス　:SYSTem:LONGform?
　　ニーモニック　LONGFORM?
　　省略形　LONG?

応答メッセージ　:SYSTem:LONGform <mode>
・「:SYSTem:LONGform」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の1桁の数字とします。
・0 が OFF、1 が ON を意味します。

3.13.5 AVERage コマンド/ AVERage? クエリ

:SYSTem:AVERage <method_arg>

本コマンドでアベレージモードの設定をします。

コマンドシンタックス　:SYSTem:AVERage <method_arg>
　　ニーモニック　AVERage
　　省略形　AVER
・<method_arg> はアベレージモード設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックとします。
・EXPonential または SUMMed のいずれかとします。

例
```
PRINT@DSO;":SYSTEM:AVERAGE EXPONENTIAL"
PRINT@DSO;":SYST:AVER EXP"
```

:SYSTem:AVERage?

本クエリでアベレージモードの設定を読み出します。

クエリシンタックス　:SYSTem:AVERage?
　　ニーモニック　AVERAGE?
　　省略形　AVER?

応答メッセージ　:SYSTem:AVERage <method>
・「:SYSTem:AVERage」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・<method> は＜文字プログラムデータ＞フォーマットとし、EXPonential または SUMMed のいずれかとします。

3.13.6 ERRor? クエリ

:SYSTem:ERRor? クエリでエラーステータスの詳細を示すコードとテキストを読み出します。

クエリシンタックス　:SYSTem:ERRor?
　　ニーモニック　ERROR?
　　省略形　ERR?

応答メッセージ　<code>,<text>
・:SYSTem:ERRor? クエリに対する応答は、:SYSTem:HEADer設定に関わらずヘッダ部を含みません。
・<code> は＜NR1 数値応答データ＞フォーマットとし、エラーコードを表す数値です。
・<text> は＜文字列応答データ＞フォーマットとし、エラーコードと 1 対 1 に対応するテキスト文字列です。
・<code>,<text> 全体で、＜任意 ASCII 応答データ＞を構成します。

エラー処理　応答メッセージが＜任意 ASCII 応答データ＞フォーマットであるため、:SYSTem:ERRor? クエリは＜プログラムメッセージ＞中の最後の＜クエリメッセージユニット＞でなくてはなりません。これに違反する場合、クエリエラーとし、以下のメッセージをエラーキューに格納します。
```
-440,"Query UNTERMINATED after indefinite response"
```

3.13.7 SET コマンド/ SET? クエリ

:SYSTem:SET <setup_arg>

本コマンドで予め読み出しておいた、本器のセットアップデータを一括して書き込みます。

コマンドシンタックス　:SYSTem:SET　#8<block_length_byte><binary_block>
ニーモニック　SET
省略形　SET(省略形なし)

・#8 は ＜プログラムデータ＞が＜任意ブロックプログラムデータ＞フォーマットであり、この後に続く<block_length_byte>が 8 桁の ASCII 数字列であることを示します。
・<block_length_byte> は 8 桁の ASCII 数字列で、この後に続く<binary_block> のバイト数を示します。リーディングゼロをサプレスせずに、必ず 8 桁の符号なし整数とします。
・<binary_block> は本器のセットアップデータをバイナリ値のブロックデータとしたものです。
・:SYSTem:SET? クエリ または *LRN? クエリで読み出しておいた＜応答データ＞をそのまま送り返して下さい。
・パラメータの省略は許されません。

エラー処理　パラメータの内容が本器用ではない場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-220,"Parameter error"

:SYSTem:SET?

本クエリでその時点の本器のセットアップデータを一括して読み出します。

クエリシンタックス　:SYSTem:SET?
ニーモニック　SET?
省略形　SET?(省略形なし)

応答メッセージ　:SYSTem:SET #8<block_length_byte><binary_block>

・「:SYSTem:SET」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
・#8 は応答メッセージが＜決定長任意ブロック応答データ＞フォーマットであり、この後に続く<block_length_byte>が 8 桁の ASCII 数字列であることを示します。
・<block_length_byte>は 8 桁の ASCII 数字列で、この後に続く<binary_block> のバイト数を示します。リーディングゼロをサプレスせずに、必ず 8 桁の符号なし整数とします。
・<binary_block> は本器のセットアップデータをバイナリ値のブロックデータとしたものです。

備　考　:SYSTem:SET? は *LRN? と同一の動作となります。

3.13.8 BALance コマンド

:SYSTem:BALance

本コマンドで本器の BALANCE 自動校正を実行します。

コマンドシンタックス　:SYSTem:BALance
　　ニーモニック　BALANCDE
　　省略形　　BAL

エラー処理　　自動校正中にエラーが発生した場合は、実行エラーとし、以下のメッセージをエラーキューに格納します。
　-240,"Hardware error"

例　　PRINT@DSO;":SYSTEM:BALANCE"
　　PRINT@DSO;":SYST:BAL"

3.13.9 GAIN コマンド

:SYSTem:GAIN

本コマンドで本器の GAIN 自動校正を実行します。

コマンドシンタックス　:SYSTem:GAIN
　　ニーモニック　GAIN
　　省略形　　GAIN（省略形なし）

エラー処理　　自動校正中にエラーが発生した場合は、実行エラーとし、以下のメッセージをエラーキューに格納します。
　-240,"Hardware error"

例　　PRINT@DSO;":SYSTEM:GAIN"
　　PRINT@DSO;":SYST:GAIN"

3.13.10 ADC コマンド

:SYSTem:ADC

本コマンドで本器の ADC 自動校正を実行します。ただし、DS-9122 では ADC 自動校正がないため本コマンドはありません。

コマンドシンタックス　:SYSTem:ADC
　　ニーモニック　ADC
　　省略形　　ADC（省略形なし）

エラー処理　　自動校正中にエラーが発生した場合は、実行エラーとし、以下のメッセージをエラーキューに格納します。
　-240,"Hardware error"

例　　PRINT@DSO;":SYSTEM:ADC"
　　PRINT@DSO;":SYST:ADC"

3.13.11 TRIG コマンド

:SYSTem:TRIG

本コマンドで本器の TRIG 自動校正を実行します。

コマンドシンタックス :SYSTem:TRIG
ニーモニック TRIG
省略形 TRIG (省略形なし)

エラー処理 自動校正中にエラーが発生した場合は、実行エラーとし、以下のメッセージをエラーキューに格納します。
-240,"Hardware error"

例
```
PRINT@DSO;":SYSTEM:TRIG"
PRINT@DSO;":SYST:TRIG"
```

3.13.12 SPEed コマンド/ SPEed? クエリ

:SYSTem:SPEed <speed_arg>

本コマンドで本器のリモートコントロールスピードを FAST モードにするか SLOW モードにするかを設定します。FAST モードにした場合、リモートコントロールのスピードは速くなりますが SHOW 表示画面の更新を行わないので注意して下さい。SLOW モードの場合はリモートコントロールスピードは遅くなりますが SHOW 表示画面は毎回更新されます。本コマンドを実行すると必ず SHOW 表示画面を更新します。FAST モードでリモートコントロール中に SHOW 表示画面を更新したい場合は本コマンドを使用してください。

コマンドシンタックス :SYSTem:SPEed
ニーモニック SPEED
省略形 SPE
・<speed_arg>はリモートコントロールスピード設定用のパラメータで、＜文字プログラムデータ＞フォーマットのニーモニックとします。
・FAST または SLOW のいずれかとします。

例
```
PRINT@DSO;":SYSTEM:SPEED FAST"
PRINT@DSO;":SYST:SPE SLOW"
```

:SYSTem:SPEed?

本クエリでリモートコントロールスピードの設定を読み出します。

コマンドシンタックス :SYSTem:SPEed?
ニーモニック SPEED?
省略形 SPE?

応答メッセージ :SYSTem:SPEed <speed>
・「:SYSTem:AVERage」部分は :SYSTem:HEADer 設定=OFF の場合には省略します。
・<speed> は＜文字プログラムデータ＞フォーマットとし、FAST または SLOW のいずれかとします。

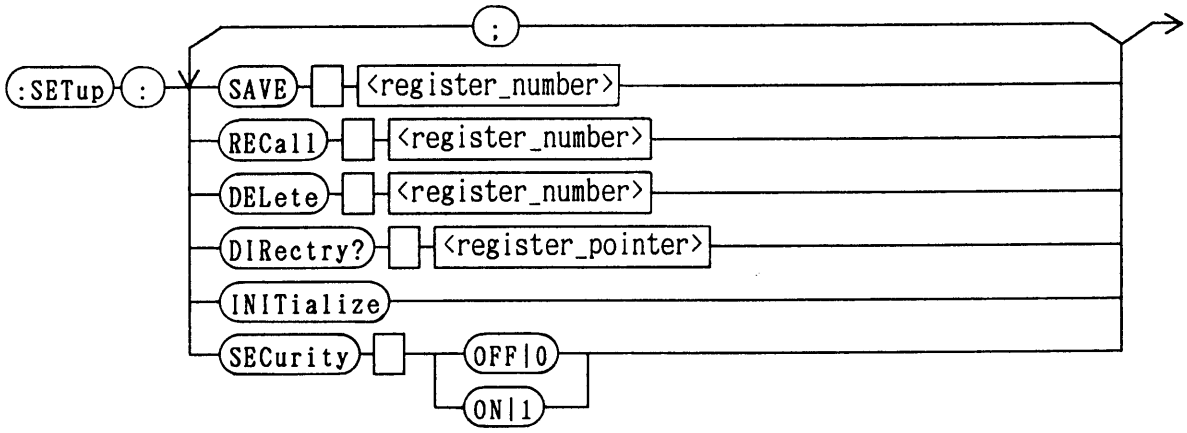
3.14 SETup 関連コマンド

:SETup:<item>…… と指定すると、本器内部メモリに対するセーブ/リコール関連の各種動作が可能です。実際の設定項目は <item> 部のヘッダにより指定されます。

コマンドシンタックス :SETup:<item>……
ニーモニック SETUP
省略形 SET

メモリカードを対象とするセーブ/リコール関連の設定、動作は :MCARd 親コマンド下に独立させます。

図3.14.1 SETup 関連コマンド全体像



3.14.1 SAVE コマンド

:SETup:SAVE <register_number_arg> コマンドで本器の内部メモリへのセットアップ情報のセーブ動作を実行します。共通コマンド *SAV と同一の機能とします。

コマンドシンタックス :SETup:SAVE <register_number_arg>
ニーモニック SAVE
省略形 SAVE(省略形なし)

・<register_number_arg> はセーブすべきファイル番号指定用パラメータで、＜数値プログラムデータ＞フォーマットの 10～98 の範囲の数値です。
・ローカル操作のセーブ/リコール動作で選択する SET010 ～ SET098 の範囲のファイル名の下 3 桁の数字を指定します。

エラー処理

・<register_number_arg> が省略された場合には、コマンドエラーとし、エラーキューに以下のメッセージを格納します。
-109,"Missing parameter"
・<register_number_arg> を丸めて整数化した値が 99 以上の場合には実行エラーとし、エラーキューに以下のメッセージを格納します。
-222,"Data out of range"
・:SETup:SECurity=ON の状態で、かつ <register_number_arg> を丸めて整数化した値によって指定される番号のセットアップファイルが既にメモリ上に存在した場合、実行エラーとし、エラーキューに以下のメッセージを格納します。
-257,"File name error"

例

```
PRINT@DSO;":SETup:SAVE 15"
PRINT@DSO;":SET:SAVE 14"
```

3.14.2 RECall コマンド

:SETup:RECall <register_number_arg> コマンドで本器の内部メモリ内に予めセーブしておいたセットアップファイルをリコールします。メモリカードにセーブしているファイルはリコールできません。共通コマンド *RCL と同一の機能とします。

コマンドシンタックス :SETup:RECall <register_number_arg>
ニーモニック RECALL
省略形 REC

・<register_number_arg> はリコールすべきファイル番号指定用パラメータで、＜数値プログラムデータ＞フォーマットの 0～98 の範囲の数値です。
・ローカル操作のセーブ/リコール動作で選択する SET010 ～ SET098 の範囲のファイル名の下 3 桁の数字を指定します。

エラー処理

・<register_number_arg> が省略された場合には、コマンド エラーとし、エラーキューに以下のメッセージを格納します。
-109,"Missing parameter"
・<register_number_arg> を丸めて整数化した値が 99 以上の場合には実行エラーとし、エラーキューに以下のメッセージを格納します。
-222,"Data out of range"
・<register_number_arg> を丸めて整数化した値によって指定される番号のセットアップ ファイルが存在しない場合、実行エラーとし、エラーキューに以下のメッセージを格納します。
-256,"File name not found"

例

```
PRINT@DSO;":SETUP:RECALL 15"
PRINT@DSO;":SET:REC 14"
```

3.14.3 DELete コマンド

:SETup:DELete <register_number_arg> コマンドで本器の内部メモリから、指定された番号のセットアップ情報ファイルを削除します。

コマンドシンタックス　:SETup:DELete <register_number>

ニーモニック　DELETE

省略形　DEL

・<register_number_arg> は削除すべきファイル番号指定用パラメータで、＜数値プログラムデータ＞フォーマットの 10～98 の範囲の数値です。

・ローカル操作のセーブ/リコール動作で選択する SET010 ～ SET098 の範囲のファイル名の下 3 桁の数字を指定します。

エラー処理

・<register_number_arg> が省略された場合には、コマンドエラーとし、以下のメッセージをエラーキューに格納します。
-109,"Missing parameter"

・<register_number_arg> を丸めて整数化した値が 99 以上の場合には実行エラーとし、エラーキューに以下のメッセージを格納します。
-222,"Data out of range"

・<register_number_arg> を丸めて整数化した値によって指定される番号のセットアップファイルが存在しない場合、実行エラーとし、エラーキューに以下のメッセージを格納します。
-256,"File name not found"

例

```
PRINT@DSO;":SETUP:DELETE 15"
PRINT@DSO;":SET:DEL 14"
```

3.14.4 DIRectry? クエリ

:SETup:DIRectry? <register_pointer_arg> クエリで本器の内部メモリから、指定された番号のセットアップ情報ファイルのディレクトリ情報を読み取ります。

クエリシンタックス　:SETup:DIRectry? <register_pointer_arg>

ニーモニック　DIRECTRY?

省略形　DIR?

・<register_pointer_arg> はファイル番号指定用パラメータで、＜数値プログラムデータ＞フォーマットの 0～98 の範囲の数値です。
・ローカル操作のセーブ/リコール動作で選択する SET000 ～ SET098 の範囲のファイル名の下 3 桁の数字を指定します。
・<register_pointer_arg> で指定された番号のファイルが存在しない場合には、指定された番号より大きい番号で一番近い番号のファイルを対象に応答データ作成を実施します。
・<register_pointer_arg> を丸めて整数化した値によって指定される番号以上で、99 番未満のセットアップファイルが存在しない場合、固定パターンの応答(備考参照)とします。
・<register_pointer_arg> は、電源投入時と :SETup:INITialize コマンド実行時とに、最小値に初期化します。
・複数ファイルに対する :SETup:DIRectry? クエリ連続実行を容易にするために、:SETup:DIRectry? クエリ実行後、そのときの <register_pointer_arg> 値に +1 した値を保持しておき、次の:SETup:DIRectry? クエリでパラメータが省略された場合に使用します。

エラー処理　<register_pointer_arg> を丸めて整数化した値が 99 以上の場合には実行エラーとし、エラーキューに以下のメッセージを格納します。
-222,"Data out of range"

例
```
PRINT@DSO;":SETUP:DIRECTRY?"
PRINT@DSO;":SET:DIR? 14"
```

応答メッセージ　<directry_data><NL^END>

・:SETup:DIRectry? クエリに対する応答データは、:SYSTem:HEADer設定に関わらずヘッダ部を含みません。
・<directry_data> は＜任意 ASCII 応答データ＞フォーマットとし、カンマで区切られた 4 つのフィールドに分かれます。
<directry_data>=<name>,<date>,<time>,<text>

<name>:
1 番目のフィールドはファイル名をあらわします。
長さは 6 文字固定で、ASCII コードの SET 3 文字と、ファイル番号を示す符号なし整数 3 桁で構成されます。ファイル番号はリーディングゼロサプレ スをしません。内部メモリ上のセットアップファイル番号は 98 以下ですので、百の位の数字は常に 0 となります。

<date>
2 番目のフィールドはファイル作成年月日をあらわします。
フォーマットは YYYY-MM-DD の 10 文字で、YYYY は 4 桁の年号、MM は 01～12 の範囲の 2 桁の数字で月を、DD は 01～31 の範囲の 2 桁の数字で日を表します。月(MM)も日(DD)もリーディングゼロサプレスをしません。:SYSTem:DATE コマンドのパラメータとはフォーマットが異なることにご注意ください。

<time>
3 番目のフィールドはファイル作成時刻をあらわします。
フォーマットは HH:MM:SS の 8 文字で、HH は 00～23 の範囲の 2 桁の数字で時を、MM は 00～59 の範囲の 2 桁の数字で分を、SS は 00～59 の範囲の 2 桁の数字で秒を表します。時(HH)も分(MM)も秒(SS)もリーディングゼロサプレスをしません。
:SYSTem:TIME コマンドのパラメータとはフォーマットが異なることにご注意下さい。
<text>
4 番目のフィールドはコメントの先頭 20 文字です。
ファイルに格納されているコメントの先頭 20 文字をそのままとし、英字の小文字は小文字のまま送信します。

エラー処理

<directry_data> が＜任意 ASCII 応答データ＞フォーマットであるため、:SETup:DIRectry? は 1 個の＜プログラムメッセージ＞中の最後の＜クエリメッセージユニット＞でなくてはならない。これに違反する場合、クエリエラーとし、:SETup:DIRectry? 以降のクエリに対する応答データの作成、送信を行いません。また、エラーキューに以下のメッセージを格納します。
-440,"Query UNTERMINATED after indefinite response"

備　考

指定番号以上の値を持つファイルが存在しない場合、以下の応答メッセージとします。
<name>=SET-00
<date>=SET000(ROM上の初期化ファイル)の作成年月日
<time>=SET000(ROM上の初期化ファイル)の作成時刻
<text>=No more file

3.14.5 INITialize コマンド

:SETup:INITialize コマンドで本器の内部メモリから、全てのセットアップ情報ファイルを削除します。ただし、SET000 を除きます。

コマンドシンタックス　:SETup:INITialize
　　ニーモニック　INITIALIZE
　　省略形　INIT

エラー処理　:SETup:SECurity設定=ON の状態で、かつ 1 個でも SET000 以外のセットアップファイルがメモリ上に存在する場合、実行エラーとし、以下のメッセージをエラーキューに格納します。
　-221,"Setting conflict"

例
```
PRINT@DSO;":SETUP:INITIALIZE"
PRINT@DSO;":SET:INIT"
```

3.14.6 SECurity コマンド/ SECurity? クエリ

:SETup:SECurity <mode_arg>

本コマンドで :SETup:SAVE コマンドと、:SETup:INITialize コマンド実行時に、同名ファイルのセーブや、ファイルが存在する時点でのイニシャライズを禁止すます。

注　　意

・本コマンドに相当するローカル操作はありません。
・本コマンドによる設定は、ローカル操作には影響しません。

コマンドシンタックス　:SETup:SECurity <mode_arg>
　　ニーモニック　SECURITY
　　省略形　　SEC
　　・<mode_arg> はセキュリティ機能 ON/OFF 設定用のパラメータで、＜文字プログラムデータ＞フォーマットの ON または OFF、＜数値プログラムデータ＞フォーマットの数値の何れかとします。
　　・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

例　　PRINT@DSO;":SETUP:SECURITY ON"
　　PRINT@DSO;":SET:SEC 0"

:SETup:SECurity?

本クエリで、現在のセキュリティの設定(ON/OFF)を読み出します。

クエリシンタックス　:SETup:SECurity?
　　ニーモニック　SECURITY?
　　省略形　　SEC?

応答メッセージ　　:SETup:SECurity <mode>
　　・「:SETup:SECurity」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の 1 桁の数字とします。
　　・0 が OFF、1 が ON を意味します。

3.15 MCARd 関連コマンド

:MCARd:<item>……　と指定すると、メモリカードに対するセーブ/リコール関連の各種動作が可能です。実際の設定項目は <item> 部のヘッダにより指定されます。

コマンドシンタックス　:MCARd:<item>……
　　ニーモニック　MCARD
　　省略形　　MCAR

エラー処理
　MCARD 親コマンド下の SAVE,RECALL などのコマンド/クエリに応じてメモリカードをアクセスしたときに共通的に起こり得るエラーについて規定します。
　以下に規定するエラーは総て実行エラーとなります。

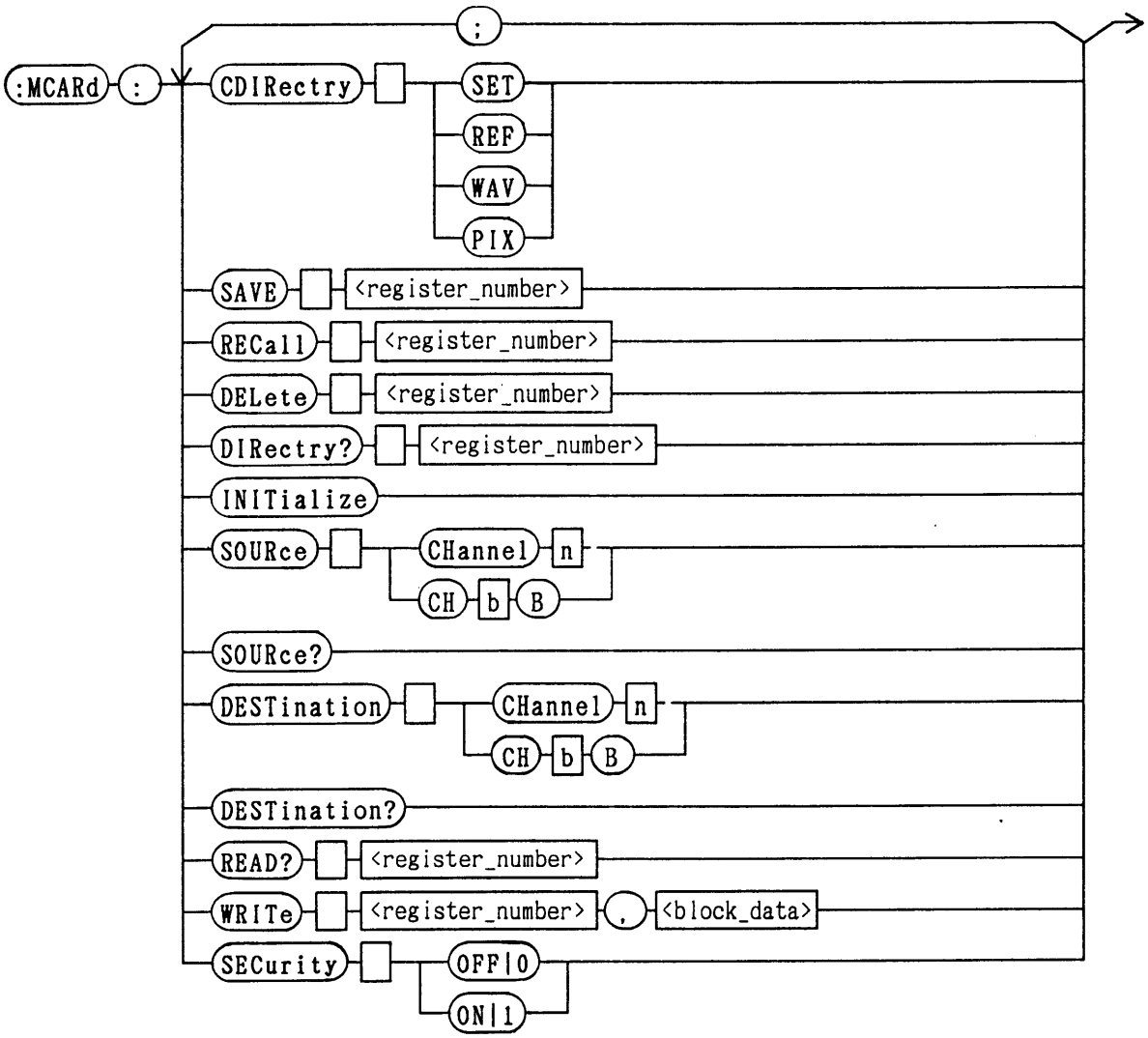
エラーキューの内容　　-252,"Missing media"
　　メモリカードが挿入されていないために、指定された動作を実行できなかった。

エラーキューの内容　　-253,"Corrupt media"
　　不正なメディア(フォーマットされていないカードである場合や指定の SRAM カードではない場合など)であるために、指定された動作を実行できなかった。

エラーキューの内容　　-254,"Media full"
　　メディアがフルであるために、指定された動作を実行できなかった。

エラーキューの内容　　-258,"Media protected"
　　メディアがプロテクトされているために、指定された動作を実行できなかった。

図3.15.1　MCARd関連コマンド全体像



3.15.1 CDIRectry コマンド/ :MCARd:CDIRectry? クエリ

:MCARd:CDIRectry <directry_arg>

本コマンドでメモリカードとの間でセーブ/リコール関連の各種動作を実施する場合の対象データ種別を切り換えます。

コマンドシンタックス　:MCARd:CDIRectry <directry_arg>
　　ニーモニック　CDIRECTRY
　　省略形　CDIR
　　・<directry_arg> はデータ種別設定用パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックとします。
　　・SETup、REF、WAVeform、PIXel の何れかとします。

エラー処理　ディスプレイモード(:DISPlay:MODE)設定≠PERSISTENCE の状態で PIXel を指定しようとした場合には、実行エラーとし、以下のメッセージをエラーキューに格納します。
　-221,"Setting conflict"

例
```
PRINT@DSO;":MCARD:CDIRECTRY SETUP"
PRINT@DSO;":MCAR:CDIR SET"
```

:MCARd:CDIRectry?

本クエリでメモリカードのセーブ/リコール用の対象データ種別を読みとります。

クエリシンタックス　:MCARd:CDIRectry?
　　ニーモニック　CDIRECTRY?
　　省略形　CDIR?

応答メッセージ　:MCARd:CDIRectry <directry>
　　・「:MCARd:CDIRectry」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<directry> は現在のデータ種別を示すパラメータで、＜文字応答データ＞フォーマットとし、SETup、REF、WAVeform、PIXel の何れかです。

3.15.2 SAVE コマンド

:MCARd:SAVE <register_number_arg> コマンドでメモリカードへファイルのセーブ動作を実行します。ファイルの種類は :MCARd:CDIRectry コマンド または セーブ/リコールメニューのローカル操作によりデータ種別として指定されたものとなります。

コマンドシンタックス　:MCARd:SAVE <register_number>
ニーモニック　SAVE
省略形　SAVE(省略形なし)

・<register_number_arg> はセーブすべきファイル番号指定用パラメータで、＜数値プログラムデータ＞フォーマットの 100～999 の範囲の数値です。
・ローカル操作のセーブ/リコール動作で選択するファイル名の下 3 桁の数字を指定します。

エラー処理

・<register_number_arg> が省略された場合には、コマンドエラーとし、エラーキューに以下のメッセージを格納します。
-109,"Missing parameter"
・ディスプレイモード(:DISPlay:MODE)設定≠PERSISTENCEかつ、:MCARd:CDIRectry設定=PIXel の状態で SAVE コマンドを実行しようとした場合には、実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"
・<register_number_arg> を丸めて整数化した値が 100 未満 または 1000 以上の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-222,"Data out of range"
・:MCARd:SECurity設定=ON の状態で、かつ <register_number_arg> を丸めて整数化した値によって指定される番号のファイルが既にメモリカード上に存在した場合、実行エラーとし、以下のメッセージをエラーキューに格納します。
-257,"File name error"

例

```
PRINT@DSO;":MCARd:SAVE 115"
PRINT@DSO;":MCAR:SAVE 114"
```

3.15.3 RECall コマンド

:MCARd:RECall <register_number_arg> コマンドでメモリカードからのファイルのリコール動作を実行します。ファイルの種類は :MCARd:CDIRectry コマンド または ローカル操作のセーブ/リコールメニューでデータ種別として指定されたものとなります。

コマンドシンタックス :MCARd:RECall <register_number_arg>
ニーモニック RECALL
省略形 REC

- <register_number_arg> はリコールすべきファイル番号指定用パラメータで、＜数値プログラムデータ＞フォーマットの 100～999 の範囲の数値です。
- ローカル操作のセーブ/リコール動作で選択するファイル名の下 3 桁の数字を指定します。

エラー処理

- <register_number_arg> が省略された場合には、コマンドエラーとし、エラーキューに以下のメッセージを格納します。
  -109,"Missing parameter"
- ディスプレイモード(:DISPlay:MODE)設定≠PERSISTENCEかつ、:MCARd:CDIRectry設定=PIXel の状態で RECall コマンドを実行しようとした場合には、実行エラーとし、以下のメッセージをエラーキューに格納します。
  -221,"Setting conflict"
- <register_number_arg> を丸めて整数化した値が 100 未満または 1000 以上の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
  -222,"Data out of range"
- <register_number_arg> を丸めて整数化した値によって指定される番号のファイルが存在しない場合、実行エラーとし、以下のメッセージをエラーキューに格納します。
  -256,"File name not found"

例

```
PRINT@DSO;":MCARD:RECALL 115"
PRINT@DSO;":MCAR:REC 114"
```

3.15.4 DELete コマンド

:MCARd:DELete <register_number_arg> コマンドでメモリカードから、指定された番号のファイルを削除します。ファイルの種類は :MCARd:CDIRectry コマンド または ローカル操作のセーブ/リコールメニューでデータ種別として指定されたものとなります。

コマンドシンタックス :MCARd:DELete <register_number_arg>
ニーモニック DELETE
省略形 DEL

- <register_number_arg> は削除すべきファイル番号指定用パラメータで、＜数値プログラムデータ＞フォーマットの 100～999 の範囲の数値です。
- ローカル操作のセーブ/リコール動作で選択するファイル名の下 3 桁の数字を指定します。

エラー処理

- <register_number_arg> が省略された場合には、コマンドエラーとし、エラーキューに以下のメッセージを格納します。
  -109,"Missing parameter"
- <register_number_arg> を丸めて整数化した値が 100 未満 または 1000 以上の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
  -222,"Data out of range"
- <register_number_arg> を丸めて整数化した値によって指定される番号のファイルが存在しない場合、実行エラーとし、以下のメッセージをエラーキューに格納します。
  -256,"File name not found"

例

```
PRINT@DSO;":MCARD:DELETE 115"
PRINT@DSO;":MCAR:DEL 115"
```

3.15.5 DIRectry? クエリ

:MCARd:DIRectry? <register_number_arg> クエリでメモリカードから、指定された番号のファイルのディレクトリ情報を読み取ります。ファイルの種類は :MCARd:CDIRectry コマンド または ローカル操作のセーブ/リコールメニューでデータ種別として指定されたものとなります。

クエリシンタックス　:MCARd:DIRectry? <register_pointer_arg>
ニーモニック　DIRECTRY?
省略形　DIR?

- <register_pointer_arg> はファイル番号指定用パラメータで、＜数値プログラムデータ＞フォーマットの 100～999 の範囲の数値です。
- ローカル操作のセーブ/リコール動作で選択するファイル名の下 3 桁の数字を指定します。
- <register_pointer_arg> で指定された番号のファイルが存在しない場合には、指定された番号より大きい番号で一番近い番号のファイルを対象に応答データ作成を実施します。
- <register_pointer_arg> を丸めて整数化した値によって指定される番号以上で 999 番未満のファイルが存在しない場合、固定パターンの応答メッセージ("備考"参照)とします。
- <register_pointer_arg> は電源投入時と :MCARd:INITialize 実行時とに、最小値(100)に初期化します。
- 複数ファイルに対する :MCARd:DIRectry? クエリ連続実行を容易にするために :MCARd:DIRectry? クエリ実行後、そのときの<register_pointer>値に +1 した値を保持しておき、次の :MCARd:DIRectry? クエリでパラメータが省略された場合に使用します。

エラー処理　<register_pointer_arg> を丸めて整数化した値が 100 未満 または 1000 以上の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-222,"Data out of range"

例
```
PRINT@DSO;":MCARD:DIRECTRY? 100"
PRINT@DSO;":MCAR:DIR?"
```

応答メッセージ　<directry_data><NL^END>

- :MCARd:DIRectry? クエリに対する応答メッセージは :SYSTem:HEADer 設定に関わらずヘッダ部を含みません。
- <directry_data> は＜任意 ASCII 応答データ＞フォーマットとし、カンマで区切られた 4 つのフィールドに分かれます。
  <directry_data>=<name>,<date>,<time>,<text>
  <name>
  1 番目のフィールドはファイル名をあらわします。
  長さは 6 文字固定で ASCII コードの SET、WAV、REF または PIX の 3 文字と、ファイル番号を示す符号なし整数 3 桁で構成されます。

<date>
2 番目のフィールドはファイル作成年月日をあらわします。
フォーマットは YYYY-MM-DD の 10 文字で YYYY は 4 桁の年号、MM は 01～12 の範囲の 2 桁 の数字で月を、DD は 01～31 の範囲の 2 桁の数字で日を表します。月(MM)も日(DD)もリーディングゼロサプレスをしません。
:SYSTem:DATA コマンドのパラメータとはフォーマットが異なることにご注意下さい。
<time>
3 番目のフィールドはファイル作成時刻をあらわします。
フォーマットは HH:MM:SS の 8 文字で HH は 00～23 の範囲の 2 桁の数字時を、MM は 00～59 の範囲の 2 桁の数字で分を、SS は 00～59 の範囲の 2 桁の数字で秒を表します。時(HH)も分(MM)も秒(SS)もリーディングゼロサプレスをしません。
:SYSTem:TIME コマンドのパラメータとはフォーマットが異なることにご注意下さい。
<text>
4 番目のフィールドはコメントの先頭 20 文字です。
ファイルに格納されているコメントの先頭 20 文字をそのままとし、英字の小文字は小文字のまま送信します。

エラー処理

<directry_data> が＜任意 ASCII 応答データ＞フォーマットであるため、:MCARd:DIRectry? は＜プログラムメッセージ＞中の最後の＜クエリメッセージユニット＞でなくてはならない。これに違反する場合、クエリエラーとし :MCARd:DIRectry? 以降のクエリに対する応答メッセージの作成、送信を行いません。また、エラーキューに以下のメッセージを格納します。
-440,"Query UNTERMINATED after indefinite response"

備　考

指定番号以上の値を持つファイルが存在しない場合、以下の応答メッセージとします。
<name>=SET-OO
<date>=SETOOO(ROM上の初期化ファイル)の作成年月日
<time>=SETOOO(ROM上の初期化ファイル)の作成時刻
<text>=No more file

3.15.6 INITialize コマンド

:MCARd:INITialize コマンドでメモリカードのフォーマッティングを行ないます。

コマンドシンタックス	:MCARd:INITialize
ニーモニック	INITIALIZE
省略形	INIT
エラー処理	:MCARd:SECurity設定=ON の状態で、かつ 1 個でもファイルがメモリカード上に存在する場合、実行エラーとし、以下のメッセージをエラーキューに格納します。 -221,"Setting conflict"
例	PRINT@DSO;":MCARD:INITIALIZE" PRINT@DSO;":MCAR:INIT"

3.15.7 SOURce コマンド/ SOURce? クエリ

:MCARd:SOURce <source_arg>

本コマンドでメモリカードの波形ファイル(WAVnnn)へ、セーブ動作時のソースチャネルを設定します。

コマンドシンタックス　:MCARd:SOURce <source_arg>
　　ニーモニック　SOURCE
　　省略形　SOUR
　　・<source_arg> は、波形セーブ用ソースチャネル設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックとします。
　　・CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B のいずれかとします。

エラー処理　CH1B、CH2B 指定は、B 掃引(:TIMebase:BSWeep)設定=OFF の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
　　-221,"Setting conflict"

例
```
PRINT@DSO;":MCARD:SOURCE CHANNEL1"
PRINT@DSO;":MCAR:SOUR CH2"
```

:MCARd:SOURce?

本クエリで、現在の波形ファイルセーブ動作用のソースチャネル設定を読み出します。

クエリシンタックス　:MCARd:SOURce?
　　ニーモニック　SOURCE?
　　省略形　SOUR?

応答メッセージ　:MCARd:SOURce <source>
　　・「:MCARd:SOURce」部分は:SYSTem:HEADer設定=OFF の場合には省略します。
　　・<source> は＜文字応答データ＞フォーマットのニーモニックとします。
　　・CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B のいずれかとします。

3.15.8 DESTination コマンド/ DESTination? クエリ

:MCARd:DESTination <destination_arg>

本コマンドでメモリカードからの波形ファイルのリコール動作時の対象トレースを設定します。

コマンドシンタックス　:MCARd:DESTination <destination_arg>
　　ニーモニック　DESTINATION
　　省略形　DEST
・<destination_arg> は、波形ファイルリコール用対象トレース設定パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックとします。
・CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B のいずれかとします。

エラー処理　CH1B、CH2B 指定は、B 掃引(TIMebase:BSWeep)設定=OFF の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例
```
PRINT@DSO;":MCARD:DESTINATION CHANNEL1"
PRINT@DSO;":MCAR:DEST CH2"
```

:MCARd:DESTination?

本クエリで、現在の波形リコール用対象トレース設定を読み出します。

クエリシンタックス　:MCARd:DESTination?
　　ニーモニック　DESTINATION?
　　省略形　DEST?

応答メッセージ　:MCARd:DESTination <destination>
・「:MCARd:DESTination」部分は:SYSTem:HEADer設定=OFF の場合には省略します。
・<destination> は＜文字応答データ＞フォーマットのニーモニックとします。
・CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B のいずれかとします。

3.15.9 READ? クエリ

:MCARd:READ? <register_number_arg> クエリでメモリカード上のファイルをコントローラに転送します。ファイルの種類は :MCARd:CDIRectry コマンド または ローカル操作のセーブ/リコールメニューでデータ種別として指定されたものとなります。

クエリシンタックス　:MCARd:READ? <register_number_arg>

ニーモニック　READ?

省略形　READ?(省略形なし)

・<register_number_arg> は読み出すべきファイル番号指定用パラメータで、＜数値プログラムデータ＞フォーマットの 100～999 の範囲の数値です。

・ローカル操作のセーブ/リコール動作で選択するファイル名の下 3 桁の数字を指定します。

エラー処理

・<register_number_arg> が省略された場合には、コマンドエラーとし、エラーキューに以下のメッセージを格納します。
-109,"Missing parameter"

・<register_number_arg> を丸めて整数化した値が 100 未満 または 1000 以上の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
-222,"Data out of range"

・<register_number_arg> を丸めて整数化した値によって指定される番号のファイルが存在しない場合、実行エラーとし、以下のメッセージをエラーキューに格納します。
-256,"File name not found"

応答メッセージ　#8<block_length_byte><binary_block>

・:MCARd:READ? クエリに対する応答メッセージは、:SYSTem:HEADer設定に関わらずヘッダ部を含みません。

・#8 は、応答メッセージが＜決定長任意ブロック応答データ＞フォーマットであり、この後に続く<block_length_byte>が 8 桁の ASCII 数字列であることを示します。

・<block_length_byte> は 8 桁の ASCII 数字列で、この後に続く<binary_block> のバイト数を示します。リーディングゼロをサプレスせずに、必ず 8 桁の符号なし整数とします。

・<binary_block> はメモリカード上のファイルデータをバイナリ値のブロックデータとしたものです。

・転送フォーマットの変更や転送バイト数の指定(分割)はできません。
また、設定されたデリミタの種類に関わりなく、デリミタは付加されずに、<binary_block> の最終バイトと同期して EOI 信号を出力します。

3.15.10 WRITe コマンド

:MCARd:WRITe <register_number_arg>,<binary_block> コマンドでコントローラからメモリカードへのファイルの書き戻しを実行します。ファイルの種類は :MCARd:CDIRectry コマンド または ローカル操作のセーブ/リコールメニューでデータ種別として指定されたものとなります。

コマンドシンタックス :MCARd:WRITe <register_number>,<block_data>
ニーモニック WRITE
省略形 WRIT

- <register_number_arg> は書き込むべきファイル番号指定用パラメータで、<数値プログラムデータ>フォーマットの 100～999 の範囲の数値です。
- ローカル操作のセーブ/リコール動作で選択するファイル名の下 3 桁の数字を指定します。
- <block_data> はメモリカード上に書き込むべきファイルデータをバイナリ値のブロックデータとしたものです。
- 基本的には、:MCARd:READ? クエリの応答メッセージをそのまま返すものとします。

エラー処理

- <register_number_arg> が省略された場合には、コマンドエラーとし、エラーキューに以下のメッセージを格納します。
  -109,"Missing parameter"
- <register_number_arg> を丸めて整数化した値が 100 未満 または 1000 以上の場合には実行エラーとし、以下のメッセージをエラーキューに格納します。
  -222,"Data out of range"
- :MCARd:SECurity設定=ON の状態で、かつ <register_number_arg> を丸めて整数化した値によって指定される番号のファイルが既にメモリカード上に存在した場合、実行エラーとし、以下のメッセージをエラーキューに格納します。
  -257,"File name error"

例

```
PRINT@DSO;":MCARd:WRITe 115"
PRINT@DSO;":MCAR:WRITe 114"
```

3.15.11 SECurity コマンド/ SECurity? クエリ

:MCARd:SECurity <mode_arg>

本コマンドで、:MCARd:SAVE コマンドと :MCARd:INITialize コマンド実行時に、同名ファイルのセーブや、ファイルが存在する時点での INITIALIZE を禁止します。

> **注　　意**
>
> ・本コマンドに相当するローカル操作はありません。
> ・本コマンドによる設定は、ローカル操作には影響しません。

コマンドシンタックス　:MCARd:SECurity <mode_arg>
　　ニーモニック　SECURITY
　　省略形　　SEC
　　・<mode_arg> はセキュリティ(ON/OFF)設定用のパラメータです。＜文字プログラムデータ＞フォーマットの ON または OFF、＜数値プログラムデータ＞フォーマットの数値の何れかとします。
　　・数値を丸めて整数化した値が 0 の場合は OFF、0 以外の場合は ON とみなします。

例　　PRINT@DSO;":MCARD:SECURITY ON"
　　PRINT@DSO;":MCAR:SEC 0"

:MCARd:SECurity?

本クエリで、現在のセキュリティの設定(ON/OFF)を読み出します。

クエリシンタックス　:MCARd:SECurity?
　　ニーモニック　SECURITY?
　　省略形　　SEC?

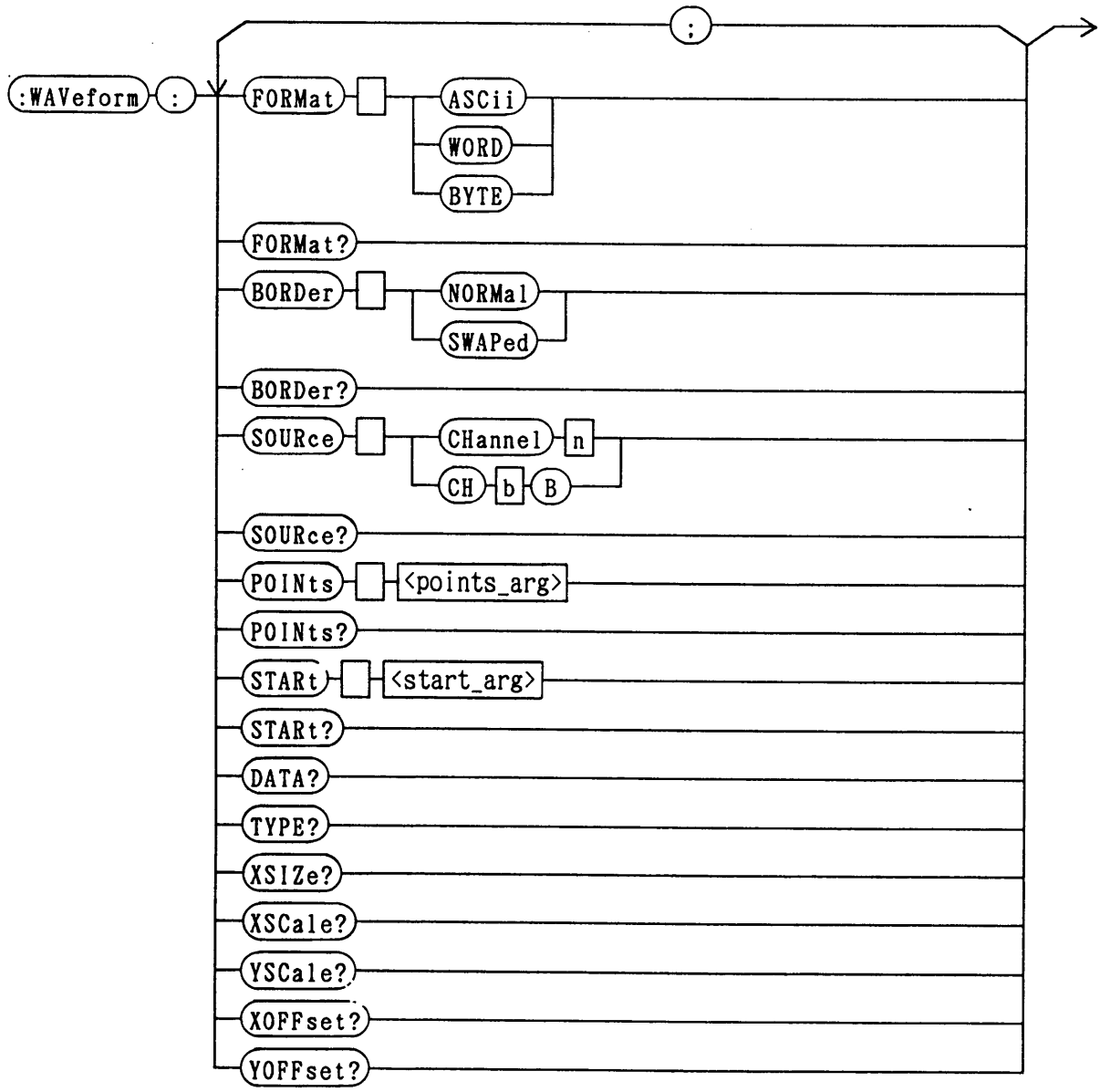
応答メッセージ　　:MCARd:SECurity <mode>
　　・「:MCARd:SECurity」部分は :SYSTem:HEADer設定=OFF の場合には省略します。
　　・<mode> は＜NR1 数値応答データ＞フォーマットとし、0 または 1 の1桁の数字とします。
　　・0 が OFF、1 が ON を意味します。

3.16 WAVeform (波形データ) 関連コマンド

:WAVeform:<item>……　と指定すると、波形データの読み出しに関わる各種設定とデータの読み出しできます。実際の設定項目は <item> 部のヘッダにより指定されます。

コマンドシンタックス :WAVeform:<item>……
ニーモニック WAVEFORM
省略形 WAV

図3.16.1 WAVeform 関連コマンド全体像



3.16.1 FORMat コマンド/ FORMat? クエリ

:WAVeform:FORMat <format_arg>

本コマンドで、:WAVeform:DATA? クエリで波形データを読み出す場合の転送フォーマットを設定します。

コマンドシンタックス　:WAVeform:FORMat <format_arg>
　　ニーモニック　FORMAT
　　省略形　FORM
・<format_arg> は、転送フォーマット設定用パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
・ASCii、WORD、BYTE のいずれかとします。

例　PRINT@DSO;":WAVEFORM:FORMAT ASCII"
　　PRINT@DSO;":WAV:FORM WORD"

:WAVeform:FORMat?

本クエリで、現在の転送フォーマットの設定値を読み出します。

クエリシンタックス　:WAVeform:FORMat?
　　ニーモニック　FORMAT?
　　省略形　FORM?

応答メッセージ　:WAVeform:FORMat <format>
・「:WAVeform:FORMat」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
・<format> は現在の転送フォーマットを示すパラメータで、＜文字応答データ＞フォーマットのニーモニックです。
・ASCii、WORD、BYTE のいずれかです。

3.16.2 BORDer コマンド/ BORDer? クエリ

:WAVeform:BORDer <byte_order_arg>

本コマンドで :WAVeform:FORMst設定=WORD として、:WAVeform:DATA? クエリで波形データを読み出す場合のバイト順序の設定をします。

コマンドシンタックス :WAVeform:BORDer <byte_order_arg>
ニーモニック BORDER
省略形 BORD

- <byte_order_arg> は、:WAVeform:FORMat設定=WORD の場合の、転送バイト順序設定用パラメータで、＜文字プログラムデータ＞フォーマットのニーモニックです。
- NORMal または SWAPed のいずれかとします。
- NORMal が上位バイト/下位バイトの順を、SWAPed が下位バイト/上位バイトの順を表します。

例

```
PRINT@DSO;":WAVEFORM:BORDER NORMAL"
PRINT@DSO;":WAV:BORD SWAP"
```

:WAVeform:BORDer?

本クエリで、現在のバイト順序の設定値を読み出します。

クエリシンタックス :WAVeform:BORDer?
ニーモニック BORDER?
省略形 BORD?

応答メッセージ :WAVeform:BORDer <byte_order>

- 「:WAVeform:BORDer」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
- <byte_order> は現在のバイト順序を示すパラメータで、＜文字応答データ＞フォーマットのニーモニックです。
- NORMal または SWAPed のいずれかとします。

3.16.3 SOURce コマンド/ SOURce? クエリ

:WAVeform:SOURce <source_arg>

本コマンドで :WAVeform:DATA? クエリで波形データを読み出す場合の、読み出しチャネルを設定します。

コマンドシンタックス :WAVeform:SOURce <source_arg>
ニーモニック SOURCE
省略形 SOUR

・<source_arg> は、読み出しチャネル設定用パラメータで、<文字プログラムデータ>フォーマットのニーモニックです。
・CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B のいずれかとします。

エラー処理

・アクイジションチャネル(:ACQuire:POINts)設定=CH1&2 となるようなメモリ長に指定されている状態で、CHannel1 または CHannel2 以外を読み出しチャネルに設定しようとした場合には実行エラーとします。
・アクイジションチャネル設定=CH1_only となるようなメモリ長に指定されている状態で、CHannel1 以外を読み出しチャネルに設定しようとした場合にも実行エラーとします。
・いずれの場合も、以下のメッセージをエラーキューに格納します。
-221,"Setting conflict"

例

```
PRINT@DSO;":WAVEFORM:SOURCE CHANNEL1"
PRINT@DSO;":WAV:SOUR CH1"
```

:WAVeform:SOURce?

本クエリで、現在の読み出しチャネルの設定値を読み出します。

クエリシンタックス :WAVeform:SOURce?
ニーモニック SOURCE?
省略形 SOUR?

応答メッセージ :WAVeform:SOURce <source>

・「:WAVeform:SOURce」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
・<source> は、現在の読み出しチャネル設定を示すパラメータで、<文字応答データ>フォーマットのニーモニックです。
・CHannel1、CHannel2、CHannel3、CHannel4、CH1B、CH2B のいずれかとします。

3.16.4 POINts コマンド/ POINts? クエリ

:WAVeform:POINts <points_arg>

本コマンドで :WAVeform:DATA? クエリで読み出される波形データの転送ワード数を設定します。

> **注　　意**
>
> :ACQuire:POINtsコマンド または ローカル操作によって、アクイジションチャネル設定を変更した場合には、:WAVeform:POINts 設定を設定可能範囲内の数値に初期化することがあります。

コマンドシンタックス　:WAVeform:POINts <points_arg>
　　ニーモニック　POINTS
　　省略形　　POIN
　　・<points_arg> は、転送ワード数設定用パラメータで、＜数値プログラムデータ＞フォーマットとします。
　　・設定範囲は 1 以上、波形データの総ワード数(メモリ長)以下とします。
　　・<points_arg> を丸めた結果と :WAVeform:STARt 設定値を組み合わせた結果が波形データの総ワード数より大きい値となってしまう場合は、波形データの総ワード数と等しくなるように、:WAVeform:STARt 設定値の方を変更します。

エラー処理　　波形データの総ワード数より大きい値を指定された場合には実行エラーとします。
　　-222,"Data out of range"

例　　PRINT@DSO;":WAVEFORM:POINTS 1024"
　　PRINT@DSO;":WAV:POIN 1e+3"

:WAVeform:POINts?

本クエリで、現在の転送ワード数の設定値を読み出します。

クエリシンタックス　:WAVeform:POINts?
　　ニーモニック　POINTS?
　　省略形　　POIN?

応答メッセージ　　:WAVeform:POINts <points>
　　・「:WAVeform:POINts」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
　　・<points> は現在の転送ワード数の設定値を示す＜NR1 数値応答データ＞フォーマットの数値です。

3.16.5 STARt コマンド/ STARt? クエリ

:WAVeform:STARt <start_arg>

本コマンドで :WAVeform:DATA? クエリで読み出される波形データの先頭番地を設定します。

> 注　　意
>
> :ACQuire:POINts コマンド または ローカル操作によって、アクイジションチャネル設定を変更した場合には、:WAVeform:STARt設定を設定可能範囲内の数値に初期化することがあります。

コマンドシンタックス　:WAVeform:STARt <start_arg>
　　ニーモニック　START
　　省略形　STAR

・<start_arg> は、転送データ先頭番地設定用パラメータで、＜数値プログラムデータ＞フォーマットとします。
・<start_arg> を丸めた結果と :WAVeform:POINts 設定値を組み合わせた結果が波形データの総ワード数より大きい値となってしまう場合は、波形データの総ワード数と等しくなるように、:WAVeform:POINts 設定値の方を変更します。

エラー処理　波形データの総ワード数より大きい値を指定された場合には実行エラーとします。
-222,"Data out of range"

例
```
PRINT@DSO;":WAVEFORM:START 0"
PRINT@DSO;":WAV:STAR 0e0"
```

:WAVeform:STARt?

本クエリで、現在の転送データ先頭番地の設定値を読み出します。

クエリシンタックス　:WAVeform:STARt?
　　ニーモニック　START?
　　省略形　STAR?

応答メッセージ　:WAVeform:STARt <start>

・「:WAVeform:STARt」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
・<start> は現在の転送開始番地の設定値を示す＜NR1 数値応答データ＞フォーマットの数値です。

3.16.6 DATA? クエリ

:WAVeform:DATA? クエリで波形データの読み出しをします。
本器が :WAVeform:DATA? クエリを受信した場合、ストップ状態とし、その時点で実施中の波形データのアクイジションを停止させます。本器から波形データの読み出しを行う場合には、基本的には、掃引方式をシングルとし、サービス要求とステータスレポート機能 または :WAIT コマンドなどを使用してアプリケーションプログラムと本器の波形データのアクイジション完了との同期をとってから、:WAVeform:DATA? クエリを使用して下さい。

注　　意

- :WAVeform:DATA? クエリによる波形データの読み出し後、波形データのアクイジションを再開させるためには、:RUN コマンドや *TRG コマンド または インタフェースメッセージ GET などによる再起動を必要とします。
- :ACQuire:POINts コマンド または ローカル操作によって、アクイジションチャネル設定を変更した場合には、:WAVeform:POINts設定と :WAVeform:STARt 設定を設定可能範囲内の数値に初期化することがあります。

クエリシンタックス　:WAVeform:DATA?
　　ニーモニック　DATA?
　　省略形　DATA?(省略形なし)

エラー処理
- :WAVeform:POINts 設定と :WAVeform:STARts 設定を組み合わせた結果が波形データの総ワード数より大きい番地となる場合、実行エラーとし、応答データを作成しません。
- アクイジションチャネル(:ACQuire:POINts)設定=CH1&2 となるようなメモリ長に指定されている状態で、かつ :WAVeform:SOURce 設定が CHannel1 でも CHannel2 でもない場合には実行エラーとします。
- アクイジションチャネル設定=CH1_only となるようなメモリ長に指定されている状態で、かつ :WAVeform:SOURce 設定が CHannel1 以外の場合にも実行エラーとします。
- いずれの場合も、以下のメッセージをエラーキューに格納します。
  -221,"Setting conflict"

応答メッセージ
- :WAVeform:DATA? クエリに対する応答メッセージは、:WAVeform:FORMat 設定(WORD、BYTE または ASCii)に応じて 3 種類となります。
- :WAVeform:SOURce 設定と :WAVeform:POINts 設定と :WAVeform:STARt 設定の影響を受けます。
- 応答メッセージは :SYSTEM:HEADER 設定に関わらずヘッダ部を含みません。

:WAVeform:FORMat設定=WORD の場合

#8<word_block_length_byte><word_binary_block>

- #8 は応答メッセージが＜決定長任意ブロック応答データ＞フォーマットであり、この後に続く <word_block_length_byte>が 8 桁の ASCII 数字列であることを示します。
- <word_block_length_byte> は 8 桁の ASCII 数字列で、この後に続く <word_ binary_block> のバイト数を示します。リーディングゼロをサプレスせずに、必ず 8 桁の符号なし整数とします。
- WORD フォーマットですので :WAVeform:POINts コマンドで指定された転送ワード数の 2倍の値となります。

・<word_binary_block> は本器内部メモリの波形データをバイナリ値のブロックデータとしたものです。
・本器の波形データは 2 バイトで 1 ワードであり、上位バイト/下位バイトの 転送順序は、:WAVeform:BORDer 設定の指定によります。

:WAVeform:FORMat設定=BYTE の場合

#8<byte_block_length_byte><byte_binary_block>

・#8 は応答メッセージが＜決定長任意ブロック応答データ＞フォーマットであり、この後に続く<word_block_length_byte>が 8 桁の ASCII 数字列であることを示します。
・<word_block_length_byte> は 8 桁の ASCII 数字列で、この後に続く<word_binary_block> のバイト数を示します。リーディングゼロをサプレスせずに、必ず 8 桁の符号なし整数とします。
・BYTE フォーマットです。:WAVeform:POINts コマンドで指定された転送ワード数と同じ値となります。
・<bite_binary_block> は本器内部メモリの波形データ上位バイト側のみをバイナリ値のブロックデータとしたものです。

:WAVeform:FORMat設定＝ASCiiの場合

<word_ascii_block><NL^END>

・<word_ascii_block> は本器内部メモリの波形データを 1 ワード(ポイント)毎に ASCII の整数値としたものを :WAVeform:POINts コマンドで指定されたワード数分だけ、カンマで区切って続けてブロックデータとしたものです。
・全体は＜任意 ASCII 応答データ＞フォーマットとなりますが、その内部は＜NR1 数値応答データ＞をカンマで区切って続けたものとなります。

エラー処理

応答メッセージがブロックデータであるため、:WAVeform:DATA? クエリは＜プログラムメッセージ＞中の最後のクエリでなくてはなりません。これに違反する場合、クエリエラーとし、:WAVeform:DATA? 以降のクエリに対する応答メッセージの作成、送信を行わず、以下のメッセージをエラーキューに格納します。

-440,"Query UNTERMINATED after indefinite response"

3.16.7 TYPE? クエリ

:WAVeform:TYPE? クエリで、現在の転送対象波形データのデータタイプを読み出します。
本器内部の波形データのデータタイプ情報の更新は、波形データのアクイジションの正常終了時となります。:WAVeform:TYPE? クエリによる読み出しも、:WAVeform:DATA? クエリと同様に、アプリケーションプログラムと本器の波形データのアクイジション完了との同期をとってからにして下さい。

クエリシンタックス　:WAVeform:TYPE?
　　ニーモニック　TYPE?
　　省略形　TYPE?(省略形なし)

応答メッセージ　:WAVeform:TYPE <type>
- 「:WAVeform:TYPE」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
- <type> は現在の転送対象波形データのデータタイプを示すパラメータで、＜文字応答データ＞フォーマットのニーモニックです。
- NORMal、AVERage、PEAK のいずれかです。
- AVERage は、波形データがアベレージング機能を使用してアクイジションされた結果であることを示します。
- PEAK は、波形データがエンベロープあるいはマックスホールド機能を使用してアクイジションされた結果であることを示します。
- NORMal は、アベレージング機能もエンベロープ機能もマックスホールド機能も使用せずにアクイジションされた結果であることを示します。

例
```
PRINT@DSO;":WAVEFORM:TYPE?"
PRINT@DSO;":WAV:TYPE?"
```

3.16.8 XSIZe? クエリ

:WAVeform:XSIZe? クエリで、現在の転送対象波形データの総波形データ長(総ワード長)を読み出します。

クエリシンタックス　:WAVeform:XSIZe?
　　ニーモニック　XSIZE?
　　省略形　XSIZ?

応答メッセージ　:WAVeform:XSIZe <data_length>
- 「:WAVeform:XSIZe」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
- <data_length> は現在の転送対象波形データの総データ長(総ポイント数)を示す、＜NR1 数値応答データ＞フォーマットの数値です。

例
```
PRINT@DSO;":WAVEFORM:XSIZE?"
PRINT@DSO;":WAV:XSIZ?"
```

3.16.9 XSCale? クエリ

:WAVeform:XSCale? クエリで現在の転送対象波形データの時間軸方向分解能(X 軸方向ドット間隔)を読み出します。

クエリシンタックス　:WAVeform:XSCale?
　　ニーモニック　XSCALE?
　　省略形　XSC?

応答メッセージ　:WAVeform:XSCale <xscale>
・「:WAVeform:XSCale」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
・<xscale> は現在の転送対象波形データのドット間の時間間隔を示すパラメータで ＜NR3 数値応答データ＞フォーマットの数値です。
・単位は s として計算しますが、サフィックス部は送信しません。

例
```
PRINT@DSO;":WAVEFORM:XSCALE?"
PRINT@DSO;":WAV:XSC?"
```

備　考　<xscale>値と<xoffset>値(:WAVeform:XOFFset?参照)を使用して、波形データのアドレスから、そのポイントのトリガ点からの時間を計算することが可能です。
time＝[data_address × <xscale>]＋<xoffset>

3.16.10 YSCale? クエリ

:WAVeform:YSCale? クエリで現在の転送対象波形データの電圧軸方向分解能を読み出します。

クエリシンタックス　:WAVeform:YSCale?
　　ニーモニック　YSCALE?
　　省略形　YSC?

応答メッセージ　:WAVeform:YSCale <yscale>
・「:WAVeform:YSCale」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
・<yscale> は現在の転送対象波形データの 1LSB 当たりの電位差を示す、＜NR3 数値応答データ＞フォーマットの数値です。
・本器の波形データは 10 ビットの A/D 変換結果を左詰めした 16 ビットデータを 1 ワードとしてますので <yscale> は波形データ収集時の電圧軸のフルスケールレンジ(電圧軸感度 VOLTS/DIV の 8 倍)を 1/65536 した値となります。:WAVeform:FORMat設定=BYTE の場合でも、WORD フォーマットを基準として計算します。
・単位は V として計算しますが、サフィックス部は送信しません。

例
```
PRINT@DSO;":WAVEFORM:YSCALE?"
PRINT@DSO;":WAV:YSC?"
```

備　考　<yscale>値と<yoffset>値(:WAVeform:YOFFset?参照)を使用して、波形データ各点の絶対値電圧を計算することができます。
voltage＝[data_code × <yscale>]＋<yoffset>

3.16.11 XOFFset? クエリ

:WAVeform:XOFFset? クエリで、現在の転送対象波形データの 0 番地のデータがトリガ点から何秒離れているかを読み出します。

クエリシンタックス　:WAVeform:XOFFset?
　　ニーモニック　XOFFSET?
　　省略形　XOFF?

応答メッセージ　:WAVeform:XOFFset <xoffset>
・「:WAVeform:XOFFset」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
・<xoffset> は現在の転送対象波形データの 0 番地データとトリガ点の時間間隔を示す＜NR3 数値応答データ＞フォーマットの数値です。
・単位は s として計算しますが、サフィックス部は送信しません。

例
```
PRINT@DSO;":WAVEFORM:XOFFSET?"
PRINT@DSO;":WAV:XOFF?"
```

備　考　<xscale>(:WAVeform:XSCale?参照)値と<xoffset>値を使用して、波形データのアドレスから、そのポイントのトリガ点からの時間を計算することが可能です。
time＝[data_address × <xscale>]＋<xoffset>

注　意　0 番地データとは全データの先頭のことであり、:WAVeform:STARt設定には依存しません。

3.16.12 YOFFset? クエリ

:WAVeform:YOFFset? クエリで、現在の転送対象波形データのコード値 0 の位置の絶対値電圧を読み出します。

クエリシンタックス　:WAVeform:YOFFset?
　　ニーモニック　YOFFSET?
　　省略形　YOFF?

応答メッセージ　:WAVeform:YOFFset <yoffset>
・「:WAVeform:YOFFset」部分は :SYSTEM:HEADER設定=OFF の場合には省略します。
・<yoffset> は現在の転送対象波形データのコード値 0 の点の絶対値電圧を示す＜NR3 数値応答データ＞フォーマットの数値です。
・単位は V として計算しますが、サフィックス部は送信しません。

例
```
PRINT@DSO;":WAVEFORM:YOFFSET?"
PRINT@DSO;":WAV:YOFF?"
```

備　考　<yscale>(:WAVeform:YSCale?参照)値と <yoffset>値を使用して、波形データ各点の絶対値電圧を計算することが可能です。
voltage＝[data_code × <yscale>]＋<yoffset>

第４部　プログラム例

4.1 GP−IB

[注]DS-9122 の場合は 1330 の FILTER を削除してください。

```
1000 '*********************************************************************
1010 '* DS-9121/22 GP-IB Sample Program                                    *
1020 '*                                                                    *
1030 '*  HARDWARE         NEC  PC-9801RA & PC-9801-29n GP-IB Interface     *
1040 '*                                                                    *
1050 '*  SOFTWARE         N-88 BASIC(86) version 6.0                       *
1060 '*                                                                    *
1070 '* Copyright (C) 1993 by IWATSU ELECTRIC CO.,LTD                      *
1080 '*                                                                    *
1090 '*********************************************************************
1100 '
1110  ISET IFC                    'interface clear
1120  ISET REN                    'remote enable
1130  DS=10                       'DS-9121/22 instrument address=10
1140  ESB=32 : TRG=1
1150  CMD DELIM=2                 '<MESSAGE TERMINATOR>=LF
1160  SRQ OFF
1170  CLS 3
1180  DIM WAVE1(1024)
1190 '
1200 '********** Status report setup *****************
1210  PRINT@ DS;"*SRE 33"         'TRG,ESB --> SRQ enable
1220  PRINT@ DS;"*ESE 61"         'OPC,QYE,DDE,EXE,CME --> ESB enable
1230  PRINT@ DS;":TESE 1"         'Trigger event --> TRG enable
1240  PRINT@ DS;"*CLS"            'Clear status
1250 '
1260  WBYTE &H14,&H3F;            'device clear
1270  ON SRQ GOSUB *INTR
1280  SRQ ON
1290 '
1300 '********** Storage setup *****************
1310  PRINT@ DS;":ACQUIRE:MEMORY SHORT;POINTS 1024;TYPE NORMAL"
1320  PRINT@ DS;":TIMEBASE:SDIV 200us;SCROLL 0;BSWEEP OFF;SAMPLING NORMAL"
1330  PRINT@ DS;":CHANNEL1:VDIV 100mV;COUPLING DC;POSITION -300mV;FILTER 0;PROBE:AUTO ON"
1340  PRINT@ DS;":TRIGGER:MODE EDGE;SWEEP SINGLE;SOURCE CH1;LEVEL 300mV"
1350  PRINT@ DS;":TRIGGER:COUPLING DC;SLOPE POSITIVE;REJECT OFF;NREJECT OFF"
1360  PRINT@ DS;":TRIGGER:POSITION 5;HOLDOFF 0s"
1370  PRINT@ DS;":DISPLAY:SPLIT OFF;MODE DOT;XY1 OFF;XY2 OFF;CALCULATE:FUNCTION OFF"
1380  PRINT@ DS;":WAVEFORM:FORMAT WORD;BORDER NORMAL;SOURCE CH1;START 0;POINTS 1024"
1390 '********** Aquisition start *****************
1400  TESR=0 :BC=0
1410  PRINT@ DS;":RUN"
1420 '********** Wait for write end ********
1430 *WEWAIT
1440  IF TESR=0 THEN GOTO *WEWAIT
1450 '
```

解　　説

1000～1090　タイトル行です。

1110～1180　各種初期設定です。

1110　このステートメントを実行すると PC-9801 は GP-IB コントローラとなります。また、システムバスにインタフェースクリアメッセージを出力します。

1120　リモートイネイブルラインを真にします。

1130　DS-9121/22 の機器アドレス 10 を変数 DS として登録します。

1140　ステータスバイトレジスタのビット値を変数として登録します。

1150　＜メッセージターミネータ＞として LF を選択します。

1160　SRQ 割り込みを禁止します。

1170　PC-9801 の画面クリアを行います。

1180　1kﾜｰﾄﾞ波形データ受信用に配列を宣言します。

1200～1280　ステータスレポート関連の初期設定を行います。

1210　DS-9121/22 のサービス要求要因として、TRG, ESBの 2 種を許可します。

1220　スタンダードイベントステータスレジスタの OPC, QYE, DDE, EXE, CME の各ビットがステータスバイトの ESB ビットに反映されることを許可します。

1230　トリガイベントステータスレジスタの内容がステータスバイトの TRG ビットに反映されることを許可します。

1240　DS-9121/22 のステータスレポート用レジスタ、キューをクリアします。

1260　デバイスクリアメッセージを出力します。

1270　SRQ 発生時の処理ルーチン（*INTR）を登録します。

1280　SRQ 割り込みを許可します。

1300～1380　DS-9121/22 のセットアップを行います。

1310　メモリ使用種別を SHORT、メモリ長を 1k ﾜｰﾄﾞ、ストレージモードを NORMAL とします。

1320　SEC/DIV を 200μs、スクロールを 0、B 掃引を OFF、サンプリングモードを NORMAL とします。

1330　CH1 の垂直軸感度を 100mV/DIV、入力結合を DC、垂直軸ポジションを -300mV、フィルタ OFF（DS-9121 のみ）、プローブ減衰比判定を AUTO にします。

[注]フィルタは DS-9121 のみです。DS-9122 の場合は削除してください。

1340　トリガモードを EDGE、掃引方式を SINGLE、信号源を CH1、トリガレベルを 300mV にします。

1350　トリガの同期結合を DC、スロープを ＋、HF/LF リジェクトを OFF、ノイズリジェクトを OFF にします。

1360　トリガポジションを 5/10、ホールドオフ時間を 0 とします。

1370　スプリット表示を OFF、表示モードを DOT、X-Y 表示を OFF、波形演算を OFF とします。

1380　波形データの転送フォーマットを WORD、バイト順序を NORMAL に、転送源を CH1、転送開始番地を 0、転送ポイント数を 1024 とします。

1390～1440　DS-9121/22 をラン状態とし、測定完了を待ちます。

1400　測定完了判定用フラグ（TESR）と転送バイトカウンタ（BC）をクリアします。

1410　DS-9121/22 をラン状態とします。

1440　測定完了を待ちます。

```
1460 '*********** CH1 WAVEFORM DATA receive *********
1470  PRINT@ DS;":WAVEFORM:DATA?"
1480  WBYTE &H3F,64+DS,32+(IEEE(1) MOD 32);     ' UNL,MTA,MLA
1490    RBYTE ;DUMMY                            ' # read
1500    RBYTE ;BL    :BL=BL-&H30                ' byte length
1510  FOR I=1 TO BL
1520    RBYTE ;BCN   :BCN=BCN-&H30
1530    BC=BC*10+BCN
1540  NEXT I
1550    WC=BC/2
1560  FOR I=0 TO WC-1
1570    RBYTE ;UPPER
1580    RBYTE ;LOWER
1590    WAVE1(I)=UPPER*256+LOWER
1600  NEXT I
1610 '
1620 '*********** Display CH1 WAVEFORM DATA *********
1630  CLS 3
1640  SCREEN 2,0
1650  WINDOW ( 0,-32768!)-(1024,32767)
1660  VIEW (63,71)-(576,329)
1670 '
1680  FOR I= 0 TO 1023
1690    YY=WAVE1(I)
1700    IF YY>32767 THEN YY=YY-65536!
1710    PSET (I,-YY)
1720  NEXT I
1730  END
1740 '
1750 '*********** SRQ INTERRUPT ROUTINE *****
1760  *INTR
1770  POLL DS,S
1780  IF (S AND ESB)<>0 THEN GOTO *ESBER
1790  IF (S AND TRG)<>0 THEN GOSUB *TRGCK
1800  SRQ ON
1810  RETURN
1820 '
1830 '*********** Error SRQ *****
1840  *ESBER
1850  PRINT@ DS;"*ESR?"
1860  INPUT@ DS;ESR
1870  PRINT "ESR=";ESR
1880  PRINT@ DS;":SYSTEM:ERROR?"
1890  INPUT@ DS;ERCODE,ERTEXT$
1900  PRINT "Error code=";ERCODE,"text=";ERTEXT$
1910  STOP
```

解　　説

1460～1600　DS-9121/22 より波形データを受信します。
1470　　DS-9121/22 に波形データ転送開始を指示します。
1480　　DS-9121/22 をトーカに、PC-9801 をリスナに指定します。
1490～1540　波形データ本体のバイト数を表す先頭部の ASCII 文字列(この例では #800002048)を受信します。
　1490　先頭の 1 文字 # を読み捨てます。
　1500　残りの ASCII 数字列を示す 1 桁の ASCII 数字文字を受信し、変数 BL とします。
　1510～1540　BL で示される桁数の ASCII 数字列を受信し、変数 BC に変換します。ここで得られた BC が波形データ本体の転送バイト数を示します。
1550　　転送バイト数 BC をワード数 WC に変換します。
1560～1600　WC で示されるワード数のバイナリ波形データを受信し、配列変数 WAVE1 に格納します。
　1570　上位バイトを受信します。
　1580　下位バイトを受信します。
　1590　上位バイトと下位バイトを 16-bit データに変換して配列変数に格納します。

1620～1730　DS-9121/22 より受信した波形データを PC-9801 の CRT に表示し、終了します。

1750～1820　SRQ 割り込み処理メイン部
　1760　SRQ 割り込み処理ルーチンラベル(*INTR)
　1770　シリアルポールを行い、DS-9121/22 が送信したステータスバイトを変数 S に格納。
　1780　ステータスバイトをチェックして、ESB ビットが ON になっていれば、処理ルーチン(*ESBER)に分岐。
　1790　ステータスバイトをチェックして、TRG ビットが ON になっていれば、処理ルーチン(*TRGCK)を呼び出し。
　1800　PC-9801 の SRQ 割り込みを再度許可。
　1810　割り込み処理から復帰。
1830～1910　スタンダードイベントステータスレジスタ内の事象に起因する SRQ の処理。
　1840　処理ルーチンのラベル(*ESBER)
　1850　スタンダードイベントステータスレジスタ読み取り用クエリを DS-9121/22 に送信。
　1860　DS-9121/22 よりの応答メッセージを受信して変数 ESR に格納。
　1870　受信した応答メッセージを PC-9801 の CRT に表示。
　1880　エラーキュー読み取り用クエリを DS-9121/22 に送信。
　1890　DS-9121/22 よりの応答メッセージを受信して、エラーコードを変数 ERCODE に、エラー内容のテキストを文字列変数 ERTEXT$ に格納。
　1900　受信した応答メッセージを PC-9801 の CRT に表示。
　1910　プログラムをストップ。

```
1920 '*********** Write end SRQ *****
1930  *TRGCK
1940  PRINT@ DS;":TESR?"
1950  INPUT@ DS;TESR
1960  RETURN
```

解　　説

1920～1960　波形測定完了に起因する SRQ の処理。
1930　処理ルーチンのラベル(*TRGCK)
1940　トリガイベントステータスレジスタ読み取り用クエリを DS-9121/22 に送信。
1950　DS-9121/22 よりの応答メッセージを受信して変数 TESR に格納。
1960　割り込み処理から復帰。

4.2 RS−232C

[注]DS-9122 の場合は 1300 の FILTER を削除してください。

```
1000 '*********************************************************************
1010 '* DS-9121/22 RS-232C Sample Program                                  *
1020 '*                                                                    *
1030 '*  HARDWARE         NEC  PC-9801NA/c                                 *
1040 '*                                                                    *
1050 '*  SOFTWARE         N-88 BASIC(86) version 6.0                       *
1060 '*                                                                    *
1070 '* Copyright (C) 1993 by IWATSU ELECTRIC CO.,LTD                      *
1080 '*                                                                    *
1090 '*********************************************************************
1100 '
1110  OPEN "COM:N81XN" AS #1
1120  ESB=32 :TRG=1
1130  CLS 3
1140  ACK$=CHR$(&H6) :NAK$=CHR$(&H15) :LF$=CHR$(&HA)
1150  DIM WAVE1(1024)
1160  CM$="RMT" :GOSUB *CMDOUT'remote enable
1170 '
1180 '********** Status report setup ****************
1190  CM$="*SRE 33":GOSUB *CMDOUT       'TRG,ESB --> SRQ enable
1200  CM$="*ESE 61":GOSUB *CMDOUT       'OPC,QYE,DDE,EXE,CME --> ESB enable
1210  CM$="TESE 1":GOSUB *CMDOUT        'Trigger event --> TRG enable
1220  CM$="*CLS":GOSUB *CMDOUT          'Clear status
1230  CM$="DCL":GOSUB *CMDOUT           'device clear
1240 '
1250 '********** Storage setup ****************
1260  CM$=":ACQUIRE:MEMORY SHORT;POINTS 1024;TYPE NORMAL"
1270  GOSUB *CMDOUT
1280  CM$=":TIMEBASE:SDIV 200us;SCROLL 0;BSWEEP OFF;SAMPLING NORMAL"
1290  GOSUB *CMDOUT
1300  CM$=":CHANNEL1:VDIV 100mV;COUPLING DC;POSITION -300mV;FILTER 0;PROBE:AUTO ON"
1310  GOSUB *CMDOUT
1320  CM$=":TRIGGER:MODE EDGE;SWEEP SINGLE;SOURCE CH1;LEVEL 300mV"
1330  GOSUB *CMDOUT
1340  CM$=":TRIGGER:COUPLING DC;SLOPE POSITIVE;REJECT OFF;NREJECT OFF"
1350  GOSUB *CMDOUT
1360  CM$=":TRIGGER:POSITION 5;HOLDOFF 0s"
1370  GOSUB *CMDOUT
1380  CM$=":DISPLAY:SPLIT OFF;MODE DOT;XY1 OFF;XY2 OFF;CALCULATE:FUNCTION OFF"
1390  GOSUB *CMDOUT
1400  CM$=":WAVEFORM:FORMAT WORD;BORDER NORMAL;SOURCE CH1;START 0;POINTS 1024"
1410  GOSUB *CMDOUT
1420 '
1430 '********** Aquisition start ****************
1440  TESR=0 :BC=0
1450  CM$=":RUN" :GOSUB *CMDOUT
1460 '
```

解　　説

本プログラムを実行するときは、DS-9121/22 側を以下のように設定してください。
・ボーレート　　：パソコン側と同じ転送レート
・パリティ　　：なし
・ストップビット：1
・デリミタ　　：CR/LF

1000～1090　タイトル行です。

1110～1160　各種初期設定です。

1110　RS-232C コミュニケーションファイルをオープンします。
また、各種パラメータを設定します。

1120　ステータスバイトレジスタのビット値を変数として登録します。

1130　PC-9801 の画面クリアを行います。

1140　ACK, NACK, LF キャラクタコードを設定します。

1150　1kﾜｰﾄﾞ波形データ受信用に配列を宣言します。

1160　機器をリモート状態にするコマンド "RMT" を送信します。

1180～1230　ステータスレポート関連の初期設定を行います。

1190　DS-9121/22 のサービス要求要因として、TRG, ESBの 2 種を許可します。

1200　スタンダードイベントステータスレジスタの OPC, QYE, DDE, EXE, CME の各ビットがステータスバイトの ESB ビットに反映されることを許可します。

1210　トリガイベントステータスレジスタの内容がステータスバイトの TRG ビットに反映されることを許可します。

1220　DS-9121/22 のステータスレポート用レジスタ、キューをクリアします。

1230　デバイスクリアメッセージを出力します。

1250～1420　DS-9121/22 のセットアップを行います。

1260　メモリ使用種別を SHORT、メモリ長を 1k ﾜｰﾄﾞ、ストレージモードを NORMAL とします。

1280　SEC/DIV を 200μs、スクロールを 0、B 掃引を OFF、サンプリングモードを NORMAL とします。

1300　CH1 の垂直軸感度を 100mV/DIV、入力結合を DC、垂直軸ポジションを -300mV、フィルタを OFF (DS-9121 のみ)、プローブ減衰比判定を AUTO にします。
[注]フィルタは DS-9121 のみです。DS-9122 の場合は削除してください。

1320　トリガモードを EDGE、掃引方式を SINGLE、信号源を CH1、トリガレベルを 300mV にします。

1340　トリガの同期結合を DC、スロープを ＋、HF/LF リジェクトを OFF、ノイズリジェクトを OFF にします。

1360　トリガポジションを 5/10、ホールドオフ時間を 0 とします。

1380　スプリット表示を OFF、表示モードを DOT、X-Y 表示を OFF、波形演算を OFF とします。

1400　波形データの転送フォーマットを WORD、バイト順序を NORMAL に、転送源を CH1、転送開始番地を 0、転送ポイント数を 1024 とします。

1430～1450　DS-9121/22 をラン状態とし、測定完了を待ちます。

1430　測定完了判定用フラグ (TESR) と転送バイトカウンタ (BC) をクリアします。

1450　DS-9121/22 をラン状態とします。

```
1470 '*********** Wait for write end ********
1480 *WEWAIT
1490  GOSUB *TRGCK
1500  IF TESR=0 THEN GOTO *WEWAIT
1510 '
1520 '*********** CH1 WAVEFORM DATA receive *********
1530  CM$=":WAVEFORM:DATA?" :GOSUB *CMDOUT
1540  DUMMY$=INPUT$(1,#1)
1550  BL=VAL(INPUT$(1,#1))
1560  FOR I=1 TO BL
1570    BCN=VAL(INPUT$(1,#1))
1580    BC=BC*10+BCN
1590  NEXT I
1600    WC=BC/2
1610  FOR I=0 TO WC-1
1620    UPPER=ASC(INPUT$(1,#1))
1630    LOWER=ASC(INPUT$(1,#1))
1640    WAVE1(I)=UPPER*256+LOWER
1650  NEXT I
1660 '
1670 '*********** Display CH1 WAVEFORM DATA *********
1680  CLS 3
1690  SCREEN 2,0
1700  WINDOW ( 0,-32768!)-(1024,32767)
1710  VIEW (63,71)-(576,329)
1720 '
1730  FOR I= 0 TO 1023
1740    YY=WAVE1(I)
1750    IF YY>32767 THEN YY=YY-65536!
1760    PSET (I,-YY)
1770  NEXT I
1780  END
1790 '
1800 '*********** Write end SRQ *****
1810  *TRGCK
1820  CM$=":TESR?" :GOSUB *CMDOUT
1830  INPUT #1,TESR
1840  RETURN
1850 '
1860 '*********** Command out *******
1870 *CMDOUT
1880  PRINT #1,CM$; :PRINT #1,LF$;
1890  LOCATE 0,0 :PRINT " < Send Command >  "
1900 *CMD100
1910   IF LOC(1)=0  THEN  *CMD100
1920   RCV$ = INPUT$(1, #1)
```

解　　説

1490　トリガイベントステータスをチェックします。
1500　測定完了を待ちます。

1520～1650　DS-9121/22 より波形データを受信します。
1530　DS-9121/22 に波形データ転送開始を指示します。
1540　先頭の 1 文字 # を読み捨てます。
1550　次の 1 文字を変数 BL とします。
1560～1590　BL で示される桁数の ASCII 数字列を受信し、変数 BC に変換します。
ここで得られた BC が波形データ本体の転送バイト数を示します。
1600　転送バイト数 BC をワード数 WC に変換します。
1610～1650　WC で示されるワード数のバイナリ波形データを受信し、配列変数 WAVE1 に格納します。

1670～1780　DS-9121/22 より受信した波形データを PC-9801 の CRT に表示し、終了します。

1800～1840　波形測定完了に起因する SRQ の処理。
1810　処理ルーチンのラベル(*TRGCK)
1820　トリガイベントステータスレジスタ読み取り用クエリを DS-9121/22 に送信。
1830　DS-9121/22 よりの応答メッセージを受信して変数 TESR に格納。
1840　割り込み処理から復帰。

1860～2030　コマンド出力、エラー処理。
1880　CM$ に格納されたコマンドを送信します。
1890　メッセージを表示します。
1910　ファイルのレコード数をチェックします。
1920　ACK または NACK のコードを受信します。

```
1930   IF RCV$=NAK$ THEN ERRFLG=1 :LOCATE 0,0 :PRINT " < NAK Receive >    "
1940   IF RCV$=ACK$ THEN ERRFLG=0 :LOCATE 0,0 :PRINT " < ACK Receive >    "
1950 ' FOR I=0 TO 5000 :NEXT I
1960   IF ERRFLG=1 THEN *CMDERR
1970 RETURN
1980 '
1990 *CMDERR
2000  CM$=":SYSTEM:ERROR?" :GOSUB *CMDOUT
2010  INPUT #1,ERCODE,ERTEXT$
2020  PRINT "Error code=";ERCODE,"text=";ERTEXT$
2030  STOP
2040 '
```

解　　説

1930　ERRFLG をセットし、メッセージを表示します。
1940　ERRFLG をリセットし、メッセージを表示します。
1960　ERRFLG がセットされていたら、エラー処理を行います。
1970　コマンド処理から復帰します(ACK を受信した場合)。
2000　エラーキュー読み取り用クエリを DS-9121/22 に送信します。
2010　DS-9121/22 よりの応答メッセージを受信して、エラーコードを変数 ERCODE に、
　　エラー内容のテキストを文字列変数 ERTEXT$ に格納。
2020　受信した応答メッセージを PC-9801 の CRT に表示。
2030　処理を中断します。

DS-9121/9122
リモート編

IWATSU